**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*AWS\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

## Determining the ELB Option That’s Best for You

Your mileage will vary depending on your exact situation, of course. But in general, the Classic Load Balancer is likely to be the best choice if your routing and load-balancing needs can all be handled based on IP addresses and TCP ports.

In contrast, the Application Load Balancer can address more complex load-balancing needs by managing traffic at the application level. This is especially advantageous for next-generation infrastructure, such as that based on containers, or if you are building complex web applications in which requests for certain components should be directed to one cluster, while others go to a different one.

**What is instance profile? How do you create a role?**

It’s a container for IAM role and you can pass this role information to a EC2-instance when the instance starts. This role gives access permissions to S3 buckets and other repositories where your applications are stored.

* We can create IAM instance profile by using AWS CLI
* Command to create instance profile is ---
* Aws iam create -instance-profile - -instance-profile-name

**What is the difference between NACL and SG?**

* NACL are applicable at the subnet level, if any instance in the subnet associated with NACL has to follow the NACL rules whereas security groups are applicable at the instance level.
* In NACL we can set both allow and deny rules for instance, whereas in SG we can only set allow rules for instance, by default everything is denied.
* SG evaluates everything before allowing the traffic. But it’s not the same case while coming to NACL. It has to first check the deny rules and the check the allow rules.

**What is cfn-init?**

It is a script that reads the template metadata from AWS::CloudFormation::init key and used to

1. Fetch data from cloud formation
2. Install packages
3. Write files to disk
4. Enable/disable, start/stop service

If you use cfn-init to update an existing file it creates a backup file with .bak extension.

**What will happen when you type aaa.com?** [[DOUBT]]]]]

* The browser extracts the domain name from the URL
* The browser queries the DNS for the IP address of the URL.
* The request sent to the DNS server is smaller than the maximum packet size, and is sent as a single packet.
* This packet also includes IP address too. This packet reaches each piece of the network equipment between the client and server, that equipment uses a routing table to figure out what node it is connected to, to be part of the fastest route to the destination.
* If that DNS server has the address of the domain then it will return it, otherwise it will forward the query along the DNS server. This happens recursively as long as the request is fulfilled.
* Assuming the DNS request is successful, the client machine now has an IP address that uniquely identifies a machine on the Internet. The web browser then assembles an HTTP request, which consists of a header and optional content. The header includes HTTP version, any relevant browser cookies etc.

**What is SSL certificate?**

When it is installed on a web server it activates the padlock and the https protocol and then make a secure connection from the web server to the browser. Mostly SSL is used to secure the credit card transactions. Usually SSL binds the domain name, server name, company’s name together. Every organization must install the SSL on its web server to securely connect to the browser.

**How to create a SSL?**

1. Generate a private key- openssl tool kit is used to generate private key and CSR. This private key is 1024bit key and is stored in pem format.

**openssl genrsa -des3 -out server.key 1024**

1. Generate a CSR- Generally this CSR is sent to Certificate Authority, who will verify the identity of the requestor and issues a certificate.

**openssl req -new -key server.key -out server.csr**

1. Remove passphrase from key- Important reason for the removal of passphrase is APACHE will ask for the passphrase every time you start the webserver.

**cp server.key server.key.org  
openssl rsa -in server.key.org -out server.key**

1. Generating a self-signed certificate- The below command creates a SSL certificate which is temporary and good for 365 days

**openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt**

1. Installing the private key and certificate-

**cp server.crt /usr/local/apache/conf/ssl.crt  
cp server.key /usr/local/apache/conf/ssl.key**

1. Configuring SSL enabled virtual hosts
2. Restart apache and Test

**I have some private servers on my premises, also I have distributed some of my workload on the public cloud, what is this architecture called?**

Hybrid Cloud

**Explanation:** This type of architecture would be a hybrid cloud. Why? Because we are using both, the public cloud, and you’re on premises servers i.e. the private cloud. To make this hybrid architecture easy to use, wouldn’t it be better if your private and public cloud were all on the same network(virtually). This is established by including your public cloud servers in a virtual private cloud, and connecting this virtual cloud with your on-premise servers using a VPN (Virtual Private Network).

**What does the following command do with respect to the Amazon EC2 security groups?**

**ec2-create-group Create Security Group**

Creates a new security group for use with your account.

**Explanation:**A Security group is just like a firewall, it controls the traffic in and out of your instance. In AWS terms, the inbound and outbound traffic. The command mentioned is pretty straight forward, it says create security group, and does the same. Moving along, once your security group is created, you can add different rules in it. For example, you have an RDS instance, to access it, you must add the public IP address of the machine from which you want access the instance in its security group.

**You have a video trans-coding application. The videos are processed according to a queue. If the processing of a video is interrupted in one instance, it is resumed in another instance. Currently there is a huge back-log of videos which needs to be processed, for this you need to add more instances, but you need these instances only until your backlog is reduced. Which of these would be an efficient way to do it?**

You should be using an **On-Demand** instance for the same. Why? First of all, the workload has to be processed now, meaning it is urgent, secondly you don’t need them once your backlog is cleared, therefore Reserved Instance is out of the picture, and since the work is urgent, you cannot stop the work on your instance just because the spot price spiked, therefore Spot Instances shall also not be used. Hence On-Demand instances shall be the right choice in this case.

**You have a distributed application that periodically processes large volumes of data across multiple Amazon EC2 Instances. The application is designed to recover gracefully from Amazon EC2 instance failures. You are required to accomplish this task in the most cost-effective way.**

**Which of the following will meet your requirements?**

Spot Instances

**Explanation:** Since the work we are addressing here is not continuous, a reserved instance shall be idle at times, same goes with On-Demand instances. Also, it does not make sense to launch an On-Demand instance whenever work comes up, since it is expensive. Hence Spot Instances will be the right fit because of their low rates and no long-term commitments.

**How is stopping and terminating an instance different from each other?**

Starting, stopping and terminating are the three states in an EC2 instance, let’s discuss them in detail:

* **Stopping and Starting** an instance: When an instance is stopped, the instance performs a normal shutdown and then transitions to a stopped state. All of its Amazon EBS volumes remain attached, and you can start the instance again at a later time. You are not charged for additional instance hours while the instance is in a stopped state.
* **Terminating** an instance: When an instance is terminated, the instance performs a normal shutdown, then the attached Amazon EBS volumes are deleted unless the volume’s *deleteOnTermination* attribute is set to false. The instance itself is also deleted, and you can’t start the instance again at a later time.

**If I want my instance to run on a single-tenant hardware, which value do I have to set the instance’s tenancy attribute to?**

Dedicated

**Explanation:** The Instance tenancy attribute should be set to Dedicated Instance. The rest of the values are invalid.

**When will you incur costs with an Elastic IP address (EIP)?**

When it is allocated and associated with a stopped instance.

**Explanation:** You are not charged, if only one Elastic IP address is attached with your running instance. But you do get charged in the following conditions:

* When you use more than one Elastic IPs with your instance.
* When your Elastic IP is attached to a stopped instance.
* When your Elastic IP is not attached to any instance.

**How is a Spot instance different from an On-Demand instance or Reserved Instance?**

First of all, let’s understand that Spot Instance, On-Demand instance and Reserved Instances are all models for pricing. Moving along, spot instances provide the ability for customers to purchase compute capacity with no upfront commitment, at hourly rates usually lower than the On-Demand rate in each region. Spot instances are just like bidding, the bidding price is called Spot Price. The Spot Price fluctuates based on supply and demand for instances, but customers will never pay more than the maximum price they have specified. If the Spot Price moves higher than a customer’s maximum price, the customer’s EC2 instance will be shut down automatically. But the reverse is not true, if the Spot prices come down again, your EC2 instance will not be launched automatically, one has to do that manually.  In Spot and On-demand instance, there is no commitment for the duration from the user side, however in reserved instances one has to stick to the time period that he has chosen.

**Are the Reserved Instances available for Multi-AZ Deployments?**

Available for all instance types

**Explanation:** Reserved Instances is a pricing model, which is available for all instance types in EC2.

**How to use the processor state control feature available on the c4.8xlarge instance?**

The processor state control consists of 2 states:

* The C state – Sleep state varying from c0 to c6. C6 being the deepest sleep state for a processor
* The P state – Performance state p0 being the highest and p15 being the lowest possible frequency.

Now, why the C state and P state. Processors have cores, these cores need thermal headroom to boost their performance. Now since all the cores are on the processor the temperature should be kept at an optimal state so that all the cores can perform at the highest performance.

Now how will these states help in that? If a core is put into sleep state it will reduce the overall temperature of the processor and hence other cores can perform better. Now the same can be synchronized with other cores, so that the processor can boost as many cores it can by timely putting other cores to sleep, and thus get an overall performance boost.

Concluding, the C and P state can be customized in some EC2 instances like the c4.8xlarge instance and thus you can customize the processor according to your workload.

**What kind of network performance parameters can you expect when you launch instances in cluster placement group?**

The network performance depends on the instance type and network performance specification, if launched in a placement group you can expect up to

* 10 Gbps in a single-flow,
* 20 Gbps in multiflow i.e., full duplex
* Network traffic outside the placement group will be limited to 5 Gbps (full duplex).

**To deploy a 4-node cluster of Hadoop in AWS which instance type can be used?**

First let’s understand what actually happens in a Hadoop cluster, the Hadoop cluster follows a master slave concept. The master machine processes all the data, slave machines store the data and act as data nodes. Since all the storage happens at the slave, a higher capacity hard disk would be recommended and since master does all the processing, a higher RAM and a much better CPU is required. Therefore, you can select the configuration of your machine depending on your workload. For e.g. – In this case c4.8xlarge will be preferred for master machine whereas for slave machine we can select i2.large instance. If you don’t want to deal with configuring your instance and installing Hadoop cluster manually, you can straight away launch an Amazon EMR (Elastic Map Reduce) instance which automatically configures the servers for you. You dump your data to be processed in S3, EMR picks it from there, processes it, and dumps it back into S3.

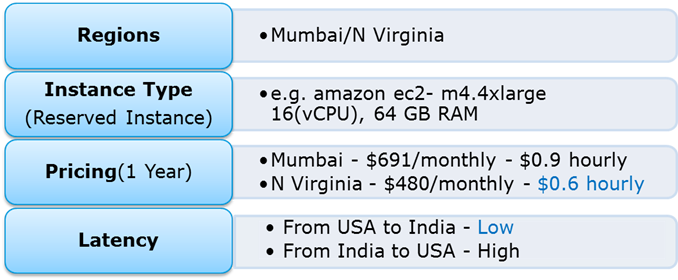
**Where do you think an AMI fits, when you are designing an architecture for a solution?**

AMIs (Amazon Machine Images) are like templates of virtual machines and an instance is derived from an AMI. AWS offers pre-baked AMIs which you can choose while you are launching an instance, some AMIs are not free, therefore can be bought from the AWS Marketplace. You can also choose to create your own custom AMI which would help you save space on AWS. For example, if you don’t need a set of software on your installation, you can customize your AMI to do that. This makes it cost efficient, since you are removing the unwanted things.

**How do you choose an Availability Zone?**

Let’s understand this through an example, consider there’s a company which has user base in India as well as in the US.

Let us see how we will choose the region for this use case :



So, with reference to the above figure the regions to choose between are, Mumbai and North Virginia. Now let us first compare the pricing, you have hourly prices, which can be converted to your per month figure. Here North Virginia emerges as a winner. But, pricing cannot be the only parameter to consider. Performance should also be kept in mind hence, let’s look at latency as well. Latency basically is the time that a server takes to respond to your requests i.e. the response time. North Virginia wins again!

So, concluding, North Virginia should be chosen for this use case.

**Is one Elastic IP address enough for every instance that I have running?**

Depends! Every instance comes with its own private and public address. The private address is associated exclusively with the instance and is returned  to Amazon EC2 only when it is stopped or terminated. Similarly, the public address is associated exclusively with the instance until it is stopped or terminated. However, this can be replaced by the Elastic IP address, which stays with the instance as long as the user doesn’t manually detach it. But what if you are hosting multiple websites on your EC2 server, in that case you may require more than one Elastic IP address.

**What are the best practices for Security in Amazon EC2?**

There are several best practices to secure Amazon EC2. A few of them are given below:

* Use AWS Identity and Access Management (IAM) to control access to your AWS resources.
* Restrict access by only allowing trusted hosts or networks to access ports on your instance.
* Review the rules in your security groups regularly, and ensure that you apply the principle of least
* Privilege – only open up permissions that you require.
* Disable password-based logins for instances launched from your AMI. Passwords can be found or cracked, and are a security risk.

**You need to configure an Amazon S3 bucket to serve static assets for your public-facing web application. Which method will ensure that all objects uploaded to the bucket are set to public read?**

Configure the bucket policy to set all objects to public read.

**Explanation:** Rather than making changes to every object, it’s better to set the policy for the whole bucket. IAM is used to give more granular permissions, since this is a website, all objects would be public by default.

**A customer wants to leverage Amazon Simple Storage Service (S3) and Amazon Glacier as part of their backup and archive infrastructure. The customer plans to use third-party software to support this integration. Which approach will limit the access of the third-party software to only the Amazon S3 bucket named “company-backup”?**

A custom IAM user policy limited to the Amazon S3 API in “company-backup”.

**Explanation:** Taking queue from the previous questions, this use case involves more granular permissions, hence IAM would be used here.

**Can S3 be used with EC2 instances, if yes, how?**

Yes, it can be used for instances with root devices backed by local instance storage. By using Amazon S3, developers have access to the same highly scalable, reliable, fast, inexpensive data storage infrastructure that Amazon uses to run its own global network of web sites. In order to execute systems in the Amazon EC2 environment, developers use the tools provided to load their Amazon Machine Images (AMIs) into Amazon S3 and to move them between Amazon S3 and Amazon EC2.

Another use case could be for websites hosted on EC2 to load their static content from S3.

**A customer implemented AWS Storage Gateway with a gateway-cached volume at their head office. An event takes the link between the main and branch office offline. Which methods will enable the branch office to access their data?**

Launch a new AWS Storage Gateway instance AMI in Amazon EC2, and restore from a gateway snapshot.

**Explanation:** The fastest way to do it would be launching a new storage gateway instance. Why? Since time is the key factor which drives every business, troubleshooting this problem will take more time. Rather than we can just restore the previous working state of the storage gateway on a new instance.

**When you need to move data over long distances using the internet, for instance across countries or continents to your Amazon S3 bucket, which method or service will you use?**

Amazon Transfer Acceleration

**Explanation:** You would not use Snowball, because for now, the snowball service does not support cross region data transfer, and since, we are transferring across countries, Snowball cannot be used. Transfer Acceleration shall be the right choice here as it throttles your data transfer with the use of optimized network paths and Amazon’s content delivery network up to 300% compared to normal data transfer speed.

**How can you speed up data transfer in Snowball?**

The data transfer can be increased in the following way:

* By performing multiple copy operations at one time i.e. if the workstation is powerful enough, you can initiate multiple cp commands each from different terminals, on the same Snowball device.
* Copying from multiple workstations to the same snowball.
* Transferring large files or by creating a batch of small file, this will reduce the encryption overhead.
* Eliminating unnecessary hops i.e. make a setup where the source machine(s) and the snowball are the only machines active on the switch being used, this can hugely improve performance.

**If you want to launch Amazon Elastic Compute Cloud (EC2) instances and assign each instance a predetermined private IP address you should:**

Launch the instances in the Amazon Virtual Private Cloud (VPC).

**Explanation:** The best way of connecting to your cloud resources (for ex- ec2 instances) from your own data center (for ex.- private cloud) is a VPC. Once you connect your datacenter to the VPC in which your instances are present, each instance is assigned a private IP address which can be accessed from your datacenter. Hence, you can access your public cloud resources, as if they were on your own network.

**Can I connect my corporate datacenter to the Amazon Cloud?**

Yes, you can do this by establishing a VPN(Virtual Private Network) connection between your company’s network and your VPC (Virtual Private Cloud), this will allow you to interact with your EC2 instances as if they were within your existing network.

**Is it possible to change the private IP addresses of an EC2 while it is running/stopped in a VPC?**

Primary private IP address is attached with the instance throughout its lifetime and cannot be changed, however secondary private addresses can be unassigned, assigned or moved between interfaces or instances at any point.

**Why do you make subnets?**

To efficiently utilize networks that have a large no. of hosts.

**Explanation:** If there is a network which has a large no. of hosts, managing all these hosts can be a tedious job. Therefore, we divide this network into subnets (sub-networks) so that managing these hosts becomes simpler.

**Which of the following is true?**

You can attach multiple subnets to a route table

**Explanation:** Route Tables are used to route network packets, therefore in a subnet having multiple route tables will lead to confusion as to where the packet has to go. Therefore, there is only one route table in a subnet, and since a route table can have any no. of records or information, hence attaching multiple subnets to a route table is possible.

**In CloudFront what happens when content is NOT present at an Edge location and a request is made to it?**

CloudFront delivers the content directly from the origin server and stores it in the cache of the edge location

**Explanation:** CloudFront is a content delivery system, which caches data to the nearest edge location from the user, to reduce latency. If data is not present at an edge location, the first time the data may get transferred from the original server, but from the next time, it will be served from the cached edge.

**If I’m using Amazon CloudFront, can I use Direct Connect to transfer objects from my own data center?**

Yes. Amazon CloudFront supports custom origins including origins from outside of AWS. With AWS Direct Connect, you will be charged with the respective data transfer rates.

**If my AWS Direct Connect fails, will I lose my connectivity?**

If a backup AWS Direct connect has been configured, in the event of a failure it will switch over to the second one. It is recommended to enable Bidirectional Forwarding Detection (BFD) when configuring your connections to ensure faster detection and failover. On the other hand, if you have configured a backup IPsec VPN connection instead, all VPC traffic will failover to the backup VPN connection automatically. Traffic to/from public resources such as Amazon S3 will be routed over the Internet. If you do not have a backup AWS Direct Connect link or a IPsec VPN link, then Amazon VPC traffic will be dropped in the event of a failure.

**If I launch a standby RDS instance, will it be in the same Availability Zone as my primary?**

No

**Explanation:** No, since the purpose of having a standby instance is to avoid an infrastructure failure (if it happens), therefore the standby instance is stored in a different availability zone, which is a physically different independent infrastructure.

**When would I prefer Provisioned IOPS over Standard RDS storage?**

If you have batch-oriented workloads

**Explanation:**  Provisioned IOPS deliver high IO rates but on the other hand it is expensive as well. Batch processing workloads do not require manual intervention they enable full utilization of systems, therefore a provisioned IOPS will be preferred for batch oriented workload.

**How is Amazon RDS, DynamoDB and Redshift different?**

* Amazon RDS is a database management service for relational databases,  it manages patching, upgrading, backing up of data etc. of databases for you without your intervention. RDS  is a Db management service for structured data only.
* DynamoDB, on the other hand, is a NoSQL database service, NoSQL deals with unstructured data.
* Redshift, is an entirely different service, it is a data warehouse product and is used in data analysis.

**If I am running my DB Instance as a Multi-AZ deployment, can I use the standby DB Instance for read or write operations along with primary DB instance?**

No

**Explanation:** No,Standby DB instance cannot be used with primary DB instance in parallel, as the former is solely used for standby purposes, it cannot be used unless the primary instance goes down.

**Your company’s branch offices are all over the world, they use a software with a multi-regional deployment on AWS, they use MySQL 5.6 for data persistence.**

**The task is to run an hourly batch process and read data from every region to compute cross-regional reports which will be distributed to all the branches. This should be done in the shortest time possible. How will you build the DB architecture in order to meet the requirements?**

For each regional deployment, use RDS MySQL with a master in the region and a read replica in the HQ region

**Explanation:** For this we will take an RDS instance as a master, because it will manage our database for us and since we have to read from every region, we’ll put a read replica of this instance in every region where the data has to be read from. It is not correct since putting a read replica would be more efficient than putting a snapshot, a read replica can be promoted if needed  to an independent DB instance, but with a Db snapshot it becomes mandatory to launch a separate DB Instance.

**Can I run more than one DB instance for Amazon RDS for free?**

Yes. You can run more than one Single-AZ Micro database instance, that too for free! However, any use exceeding 750 instance hours, across all Amazon RDS Single-AZ Micro DB instances, across all eligible database engines and regions, will be billed at standard Amazon RDS prices. For example: if you run two Single-AZ Micro DB instances for 400 hours each in a single month, you will accumulate 800 instance hours of usage, of which 750 hours will be free. You will be billed for the remaining 50 hours at the standard Amazon RDS price.

**Which AWS services will you use to collect and process e-commerce data for near real-time analysis?**

Amazon DynamoDB and Amazon Redshift

**Explanation:** DynamoDB is a fully managed NoSQL database service. DynamoDB, therefore can be fed any type of unstructured data, which can be data from e-commerce websites as well, and later, an analysis can be done on them using Amazon Redshift. We are not using Elastic MapReduce, since a near real time analyses is needed.

**Can I retrieve only a specific element of the data, if I have a nested JSON data in DynamoDB?**

Yes. When using the GetItem, BatchGetItem, Query or Scan APIs, you can define a Projection Expression to determine which attributes should be retrieved from the table. Those attributes can include scalars, sets, or elements of a JSON document.

**A company is deploying a new two-tier web application in AWS. The company has limited staff and requires high availability, and the application requires complex queries and table joins. Which configuration provides the solution for the company’s requirements?**

Amazon DynamoDB

**Explanation:** DynamoDB has the ability to scale more than RDS or any other relational database service, therefore DynamoDB would be the apt choice.

**What happens to my backups and DB Snapshots if I delete my DB Instance?**

When you delete a DB instance, you have an option of creating a final DB snapshot, if you do that you can restore your database from that snapshot. RDS retains this user-created DB snapshot along with all other manually created DB snapshots after the instance is deleted, also automated backups are deleted and only manually created DB Snapshots are retained.

**Which of the following use cases are suitable for Amazon DynamoDB? Choose 2 answers**

Storing metadata for Amazon S3 objects and Running relational joins and complex updates.

**Explanation:** If all your JSON data have the same fields ex: [id, name, age] then it would be better to store it in a relational database, the metadata on the other hand is unstructured, also running relational joins or complex updates would work on DynamoDB as well.

**How can I load my data to Amazon Redshift from different data sources like Amazon RDS, Amazon DynamoDB and Amazon EC2?**

You can load the data in the following two ways:

* You can use the COPY command to load data in parallel directly to Amazon Redshift from Amazon EMR, Amazon DynamoDB, or any SSH-enabled host.
* AWS Data Pipeline provides a high performance, reliable, fault tolerant solution to load data from a variety of AWS data sources. You can use AWS Data Pipeline to specify the data source, desired data transformations, and then execute a pre-written import script to load your data into Amazon Redshift.

**Your application has to retrieve data from your user’s mobile every 5 minutes and the data is stored in DynamoDB, later every day at a particular time the data is extracted into S3 on a per user basis and then your application is later used to visualize the data to the user. You are asked to optimize the architecture of the backend system to lower cost, what would you recommend?**

Introduce Amazon Elasticache to cache reads from the Amazon DynamoDB table and reduce provisioned read throughput.

**Explanation:** Since our work requires the data to be extracted and analyzed, to optimize this process a person would use provisioned IO, but since it is expensive, using a ElastiCache memoryinsread to cache the results in the memory can reduce the provisioned read throughput and hence reduce cost without affecting the performance.

**You are running a website on EC2 instances deployed across multiple Availability Zones with a Multi-AZ RDS MySQL Extra Large DB Instance. The site performs a high number of small reads and writes per second and relies on an eventual consistency model. After comprehensive tests, you discover that there is read contention on RDS MySQL. Which are the best approaches to meet these requirements? (Choose 2 answers)**

Deploy ElastiCache in-memory cache running in each availability zone

Increase the RDS MySQL Instance size and Implement provisioned IOPS

**Explanation:**Since it does a lot of read writes, provisioned IO may become expensive. But we need high performance as well, therefore the data can be cached using ElastiCache which can be used for frequently reading the data. As for RDS since read contention is happening, the instance size should be increased and provisioned IO should be introduced to increase the performance.

**A startup is running a pilot deployment of around 100 sensors to measure street noise and air quality in urban areas for 3 months. It was noted that every month around 4GB of sensor data is generated. The company uses a load balanced auto scaled layer of EC2 instances and a RDS database with 500 GB standard storage. The pilot was a success and now they want to deploy at least  100K sensors which need to be supported by the backend. You need to store the data for at least 2 years to analyze it. Which setup of the following would you prefer?**

Replace the RDS instance with a 6 node Redshift cluster with 96TB of storage

**Explanation:** A Redshift cluster would be preferred because it easy to scale, also the work would be done in parallel through the nodes, therefore is perfect for a bigger workload like our use case. Since each month 4 GB of data is generated, therefore in 2 years, it should be around 96 GB. And since the servers will be increased to 100K in number, 96 GB will approximately become 96TB. Hence option C is the right answer.

**Suppose you have an application where you have to render images and also do some general computing. From the following  services which service will best fit your need?**

Application Load Balancer

**Explanation:** You will choose an application load balancer, since it supports path based routing, which means it can take decisions based on the URL, therefore if your task needs image rendering it will route it to a different instance, and for general computing it will route it to a different instance.

**What is the difference between Scalability and Elasticity?**

Scalability is the ability of a system to increase its hardware resources to handle the increase in demand. It can be done by increasing the hardware specifications or increasing the processing nodes.

Elasticity is the ability of a system to handle increase in the workload by adding additional hardware resources when the demand increases(same as scaling) but also rolling back the scaled resources, when the resources are no longer needed. This is particularly helpful in Cloud environments, where a pay per use model is followed.

**How will you change the instance type for instances which are running in your application tier and are using Auto Scaling. Where will you change it from the following areas?**

Auto Scaling launch configuration

**Explanation:** Auto scaling tags configuration, is used to attach metadata to your instances, to change the instance type you have to use auto scaling launch configuration.

**You have a content management system running on an Amazon EC2 instance that is approaching 100% CPU utilization. Which option will reduce load on the Amazon EC2 instance?**

Create a load balancer, and register the Amazon EC2 instance with it

**Explanation:** Creating alone an auto-scaling group will not solve the issue, until you attach a load balancer to it. Once you attach a load balancer to an auto-scaling group, it will efficiently distribute the load among all the instances. Option B – CloudFront is a CDN, it is a data transfer tool therefore will not help reduce load on the EC2 instance. Similarly, the other option – Launch configuration is a template for configuration which has no connection with reducing loads.

**When should I use a Classic Load Balancer and when should I use an Application load balancer?**

A Classic Load Balancer is ideal for simple load balancing of traffic across multiple EC2 instances, while an Application Load Balancer is ideal for microservices or container-based architectures where there is a need to route traffic to multiple services or load balance across multiple ports on the same EC2 instance.

**What does Connection draining do?**

 Re-routes traffic from instances which are to be updated or failed a health check.

**Explanation:** Connection draining is a service under ELB which constantly monitors the health of the instances. If any instance fails a health check or if any instance has to be patched with a software update, it  pulls all the traffic from that instance and re-routes them to other instances.

**When an instance is unhealthy, it is terminated and replaced with a new one, which of the following services does that?**

Fault Tolerance

**Explanation:** When ELB detects that an instance is unhealthy, it starts routing incoming traffic to other healthy instances in the region. If all the instances in a region becomes unhealthy, and if you have instances in some other availability zone/region, your traffic is directed to them. Once your instances become healthy again, they are re-routed back to the original instances.

**What are lifecycle hooks used for in Auto-Scaling?**

 They are used to put an additional wait time to a scale in or scale out event.

**Explanation:** Lifecycle hooks are used for putting wait time before any lifecycle action i.e., launching or terminating an instance happens. The purpose of this wait time, can be anything from extracting log files before terminating an instance or installing the necessary software’s in an instance before launching it.

**A user has setup an Auto Scaling group. Due to some issue, the group has failed to launch a single instance for more than 24 hours. What will happen to Auto Scaling in this condition?**

Auto Scaling will suspend the scaling process

**Explanation:** Auto Scaling allows you to suspend and then resume one or more of the Auto Scaling processes in your Auto Scaling group. This can be very useful when you want to investigate a configuration problem or other issue with your web application, and then make changes to your application, without triggering the Auto Scaling process.

**You have an EC2 Security Group with several running EC2 instances. You changed the Security Group rules to allow inbound traffic on a new port and protocol, and then launched several new instances in the same Security Group. The new rules apply:**

Immediately to all instances in the security group.

**Explanation:** Any rule specified in an EC2 Security Group applies immediately to all the instances, irrespective of when they are launched before or after adding a rule.

**To create a mirror image of your environment in another region for disaster recovery, which of the following AWS resources do not need to be recreated in the second region? ( Choose 2 answers )**

Route 53 Record Sets and Elastic IP Addresses (EIP)

**Explanation:** Elastic IPs and Route 53 record sets are common assets therefore there is no need to replicate them, since Elastic IPs and Route 53 are valid across regions

**A customer wants to capture all client connection information from his load balancer at an interval of 5 minutes, which of the following options should he choose for his application?**

Enable AWS CloudTrail for the load-balancer.

**Explanation:** AWS CloudTrail provides inexpensive logging information for load balancer and other AWS resources. This logging information can be used for analyses and other administrative work, therefore is perfect for this use case.

**A customer wants to track access to their Amazon Simple Storage Service (S3) buckets and also use this information for their internal security and access audits. Which of the following will meet the Customer requirement?**

Enable AWS CloudTrail to audit all Amazon S3 bucket access.

**Explanation:** AWS CloudTrail has been designed for logging and tracking API calls. Also, this service is available for storage, therefore should be used in this use case.

**Which of the following are true regarding AWS CloudTrail? (Choose 2 answers)**

CloudTrail is enabled on a per-region and service basis

Logs can be delivered to a single Amazon S3 bucket for aggregation.

**Explanation:** Cloudtrail is not enabled for all the services and is also not available for all the regions. Therefore option B is correct, also the logs can be delivered to your S3 bucket, hence C is also correct.

**What happens if CloudTrail is turned on for my account but my Amazon S3 bucket is not configured with the correct policy?**

CloudTrail files are delivered according to S3 bucket policies. If the bucket is not configured or is misconfigured, CloudTrail might not be able to deliver the log files.

**How do I transfer my existing domain name registration to Amazon Route 53 without disrupting my existing web traffic?**

You will need to get a list of the DNS record data for your domain name first, it is generally available in the form of a “zone file” that you can get from your existing DNS provider. Once you receive the DNS record data, you can use Route 53’s Management Console or simple web-services interface to create a hosted zone that will store your DNS records for your domain name and follow its transfer process. It also includes steps such as updating the nameservers for your domain name to the ones associated with your hosted zone. For completing the process, you have to contact the registrar with whom you registered your domain name and follow the transfer process. As soon as your registrar propagates the new name server delegations, your DNS queries will start to get answered.

**Which of the following services you would not use to deploy an app?**

Lambda

**Explanation:** Lambda is used for running server-less applications. It can be used to deploy functions triggered by events. When we say server-less, we mean without you worrying about the computing resources running in the background. It is not designed for creating applications which are publicly accessed.

**How does Elastic Beanstalk apply updates?**

By having a duplicate ready with updates before swapping.

**Explanation:** Elastic Beanstalk prepares a duplicate copy of the instance, before updating the original instance, and routes your traffic to the duplicate instance, so that, in case your updated application fails, it will switch back to the original instance, and there will be no downtime experienced by the users who are using your application.

**How is AWS Elastic Beanstalk different than AWS OpsWorks?**

AWS Elastic Beanstalk is an application management platform while OpsWorks is a configuration management platform. BeanStalk is an easy to use service which is used for deploying and scaling web applications developed with Java, .Net, PHP, Node.js, Python, Ruby, Go and Docker. Customers upload their code and Elastic Beanstalk automatically handles the deployment. The application will be ready to use without any infrastructure or resource configuration.

In contrast, AWS Opsworks is an integrated configuration management platform for IT administrators or DevOps engineers who want a high degree of customization and control over operations.

**What happens if my application stops responding to requests in beanstalk?**

AWS Beanstalk applications have a system in place for avoiding failures in the underlying infrastructure. If an Amazon EC2 instance fails for any reason, Beanstalk will use Auto Scaling to automatically launch a new instance. Beanstalk can also detect if your application is not responding on the custom link, even though the infrastructure appears healthy, it will be logged as an environmental event ( ex: a bad version was deployed) so you can take an appropriate action.

**How is AWS OpsWorks different than AWS CloudFormation?**

OpsWorks and CloudFormation both support application modelling, deployment, configuration, management and related activities. Both support a wide variety of architectural patterns, from simple web applications to highly complex applications. AWS OpsWorks and AWS CloudFormation differ in abstraction level and areas of focus.

AWS CloudFormation is a building block service which enables customer to manage almost any AWS resource via JSON-based domain specific language. It provides foundational capabilities for the full breadth of AWS, without prescribing a particular model for development and operations. Customers define templates and use them to provision and manage AWS resources, operating systems and application code.

In contrast, AWS OpsWorks is a higher-level service that focuses on providing highly productive and reliable DevOps experiences for IT administrators and ops-minded developers. To do this, AWS OpsWorks employs a configuration management model based on concepts such as stacks and layers, and provides integrated experiences for key activities like deployment, monitoring, auto-scaling, and automation. Compared to AWS CloudFormation, AWS OpsWorks supports a narrower range of application-oriented AWS resource types including Amazon EC2 instances, Amazon EBS volumes, Elastic IPs, and Amazon CloudWatch metrics.

**I created a key in Oregon region to encrypt my data in North Virginia region for security purposes. I added two users to the key and an external AWS account. I wanted to encrypt an object in S3, so when I tried, the key that I just created was not listed.  What could be the reason?**

The Key should be in the same region.

**Explanation:** The key created and the data to be encrypted should be in the same region. Hence the approach taken here to secure the data is incorrect.

**A company needs to monitor the read and write IOPS for their AWS MySQL RDS instance and send real-time alerts to their operations team. Which AWS services can accomplish this?**

Amazon CloudWatch

**Explanation:** Amazon CloudWatch is a cloud monitoring tool and hence this is the right service for the mentioned use case. The other options listed here are used for other purposes for example route 53 is used for DNS services, therefore CloudWatch will be the apt choice.

**What happens when one of the resources in a stack cannot be created successfully in AWS OpsWorks?**

When an event like this occurs, the “automatic rollback on error” feature is enabled, which causes all the AWS resources which were created successfully till the point where the error occurred to be deleted. This is helpful since it does not leave behind any erroneous data, it ensures the fact that stacks are either created fully or not created at all. It is useful in events where you may accidentally exceed your limit of the no. of Elastic IP addresses or maybe you may not have access to an EC2 AMI that you are trying to run etc.

**What automation tools can you use to spinup servers?**

Any of the following tools can be used:

* Roll-your-own scripts, and use the AWS API tools.  Such scripts could be written in bash, Perl or other language of your choice.
* Use a configuration management and provisioning tool like puppet or its successor Opscode Chef.  You can also use a tool like Scalr.
* Use a managed solution such as Rightscale.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*DevOps\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**What is the need for DevOps?**

According to me, this answer should start by explaining the general market trend. Instead of releasing big sets of features, companies are trying to see if small features can be transported to their customers through a series of release trains. This has many advantages like quick feedback from customers, better quality of software etc. which in turn leads to high customer satisfaction. To achieve this, companies are required to:

1. Increase deployment frequency
2. Lower failure rate of new releases
3. Shortened lead time between fixes
4. Faster mean time to recovery in the event of new release crashing

DevOps fulfills all these requirements and helps in achieving seamless software delivery. You can give examples of companies like Etsy, Google and Amazon which have adopted DevOps to achieve levels of performance that were unthinkable even five years ago. They are doing tens, hundreds or even thousands of code deployments per day while delivering world class stability, reliability and security.

**How is DevOps different from Agile / SDLC?**

I would advise you to go with the below explanation:

Agile is a set of values and principles about how to produce i.e. develop software. Example: if you have some ideas and you want to turn those ideas into working software, you can use the Agile values and principles as a way to do that. But, that software might only be working on a developer’s laptop or in a test environment. You want a way to quickly, easily and repeatedly move that software into production infrastructure, in a safe and uncomplicated way. To do that you need DevOps tools and techniques.

You can summarize by saying Agile software development methodology focuses on the development of software but DevOps on the other hand is responsible for development as well as deployment of the software in the safest and most reliable way possible.

Now remember, you have included DevOps tools in your previous answer so be prepared to answer some questions related to that.

**Which are the top DevOps tools? Which tools have you worked on?**

The most popular DevOps tools are mentioned below:

* Git : Version Control System tool
* Jenkins : Continuous Integration tool
* Selenium : Continuous Testing tool
* Puppet, Chef, Ansible : Configuration Management and Deployment tools
* Nagios : Continuous Monitoring tool
* Docker : Containerization tool

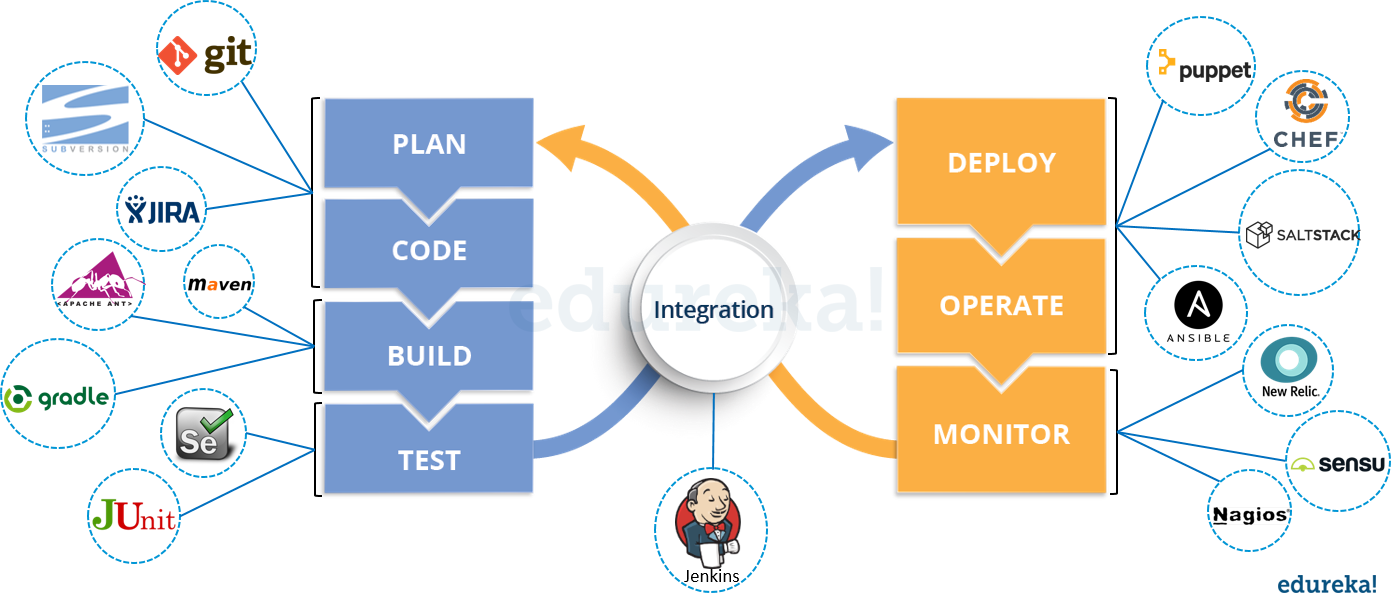
You can also mention any other tool if you want, but make sure you include the above tools in your answer.  
The second part of the answer has two possibilities:

1. If you have experience with all the above tools then you can say that I have worked on all these tools for developing good quality software and deploying those software’s easily, frequently, and reliably.
2. If you have experience only with some of the above tools then mention those tools and say that I have specialization in these tools and have an overview about the rest of the tools.

**How do all these tools work together?**

Given below is a generic logical flow where everything gets automated for seamless delivery. However, this flow may vary from organization to organization as per the requirement.

1. Developers develop the code and this source code is managed by Version Control System tools like Git etc.
2. Developers send this code to the Git repository and any changes made in the code is committed to this Repository.
3. Jenkins pulls this code from the repository using the Git plugin and build it using tools like Ant or Maven.
4. Configuration management tools like puppet deploys & provisions testing environment and then Jenkins releases this code on the test environment on which testing is done using tools like selenium.
5. Once the code is tested, Jenkins send it for deployment on the production server (even production server is provisioned & maintained by tools like puppet).
6. After deployment, It is continuously monitored by tools like Nagios.
7. Docker containers provides testing environment to test the build features.

**What are the advantages of DevOps?**

Technical benefits:

* Continuous software delivery
* Less complex problems to fix
* Faster resolution of problems

Business benefits:

* Faster delivery of features
* More stable operating environments
* More time available to add value (rather than fix/maintain)

**What is the most important thing DevOps helps us achieve?**

According to me, the most important thing that DevOps helps us achieve is to get the changes into production as quickly as possible while minimizing risks in software quality assurance and compliance. This is the primary objective of DevOps.

However, you can add many other positive effects of DevOps. For example, clearer communication and better working relationships between teams i.e. both the Ops team and Dev team collaborate to deliver superior quality software which in turn leads to higher customer satisfaction.

**Explain with a use case where DevOps can be used in industry / real-life.**

There are many industries that are using DevOps so you can mention any of those use cases, you can also refer the below example:

Etsy is a peer-to-peer e-commerce website focused on handmade or vintage items and supplies, as well as unique factory-manufactured items. Etsy struggled with slow, painful site updates that frequently caused the site to go down. It affected sales for millions of Etsy’s users who sold goods through online market place and risked driving them to the competitor.

With the help of a new technical management team, Etsy transitioned from its waterfall model, which produced four-hour full-site deployments twice weekly, to a more agile approach. Today, it has a fully automated deployment pipeline, and its continuous delivery practices have reportedly resulted in more than 50 deployments a day with fewer disruptions.

**Explain your understanding and expertise on both the software development side and the technical operations side of an organization you have worked with in the past.**

For this answer, share your past experience and try to explain how flexible you were in your previous job. You can refer the below example:

DevOps engineers almost always work in a 24/7 business-critical online environment. I was adaptable to on-call duties and was available to take up real-time, live-system responsibility. I successfully automated processes to support continuous software deployments. I have experience with public/private clouds, tools like Chef or Puppet, scripting and automation with tools like Python and PHP, and a background in Agile.

**What are the anti-patterns of DevOps?**

A pattern is common usage usually followed. If a pattern commonly adopted by others does not work for your organization and you continue to blindly follow it, you are essentially adopting an anti-pattern. There are myths about DevOps. Some of them include:

* DevOps is a process
* Agile equals DevOps?
* We need a separate DevOps group
* DevOps will solve all our problems
* DevOps means Developers Managing Production
* DevOps is Development-driven release management
  1. DevOps is not development driven.
  2. DevOps is not IT Operations driven.
* We can’t do DevOps – We’re Unique
* We can’t do DevOps – We’ve got the wrong people

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Version Control – Git\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**What is Version control?**

This is probably the easiest question you will face in the interview. My suggestion is to first give a definition of Version control. It is a system that records changes to a file or set of files over time so that you can recall specific versions later. Version control systems consist of a central shared repository where teammates can commit changes to a file or set of files. Then you can mention the uses of version control.

Version control allows you to:

* Revert files back to a previous state.
* Revert the entire project back to a previous state.
* Compare changes over time.
* See who last modified something that might be causing a problem.
* Who introduced an issue and when?

**What are the benefits of using version control?**

I will suggest you to include the following advantages of version control:

1. With Version Control System (VCS), all the team members are allowed to work freely on any file at any time. VCS will later allow you to merge all the changes into a common version.
2. All the past versions and variants are neatly packed up inside the VCS. When you need it, you can request any version at any time and you’ll have a snapshot of the complete project right at hand.
3. Every time you save a latest version of your project, your VCS requires you to provide a little description of what was changed. Additionally, you can see what exactly was changed in the file’s content. This allows you to know who has made what change in the project.
4. A distributed VCS like Git allows all the team members to have complete history of the project so if there is a breakdown in the central server you can use any of your teammate’s local Git repository.

**Describe branching strategies you have used.**

This question is asked to test your branching experience so tell them about how you have used branching in your previous job and what purpose does it serves, you can refer the below points:

* Feature branching

A feature branch model keeps all the changes for a specific feature inside of a branch. When the feature is fully tested and validated by automated tests, the branch is then merged into master.

* Task branching

In this model, each task is implemented on its own branch with the task key included in the branch name. It is easy to see which code implements which task, just look for the task key in the branch name.

* Release branching

Once the develop branch has acquired enough features for a release, you can clone that branch to form a Release branch. Creating this branch starts the next release cycle, so no new features can be added after this point, only bug fixes, documentation generation, and other release-oriented tasks should go in this branch. Once it is ready to ship, the release gets merged into master and tagged with a version number. In addition, it should be merged back into develop branch, which may have progressed since the release was initiated.

In the end tell them that branching strategies varies from one organization to another, so I know basic branching operations like delete, merge, checking out a branch etc.

**Which VCS tool you are comfortable with?**

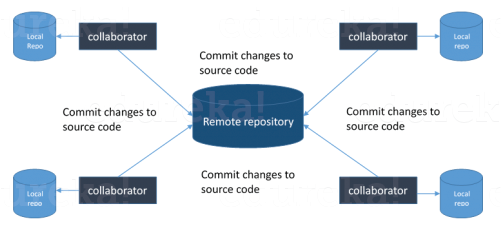
You can just mention the VCS tool that you have worked on like this: “I have worked on Git and one major advantage it has over other VCS tools like SVN is that it is a distributed version control system.”

Distributed VCS tools do not necessarily rely on a central server to store all the versions of a project’s files. Instead, every developer “clones” a copy of a repository and has the full history of the project on their own hard drive.

**What is Git?**

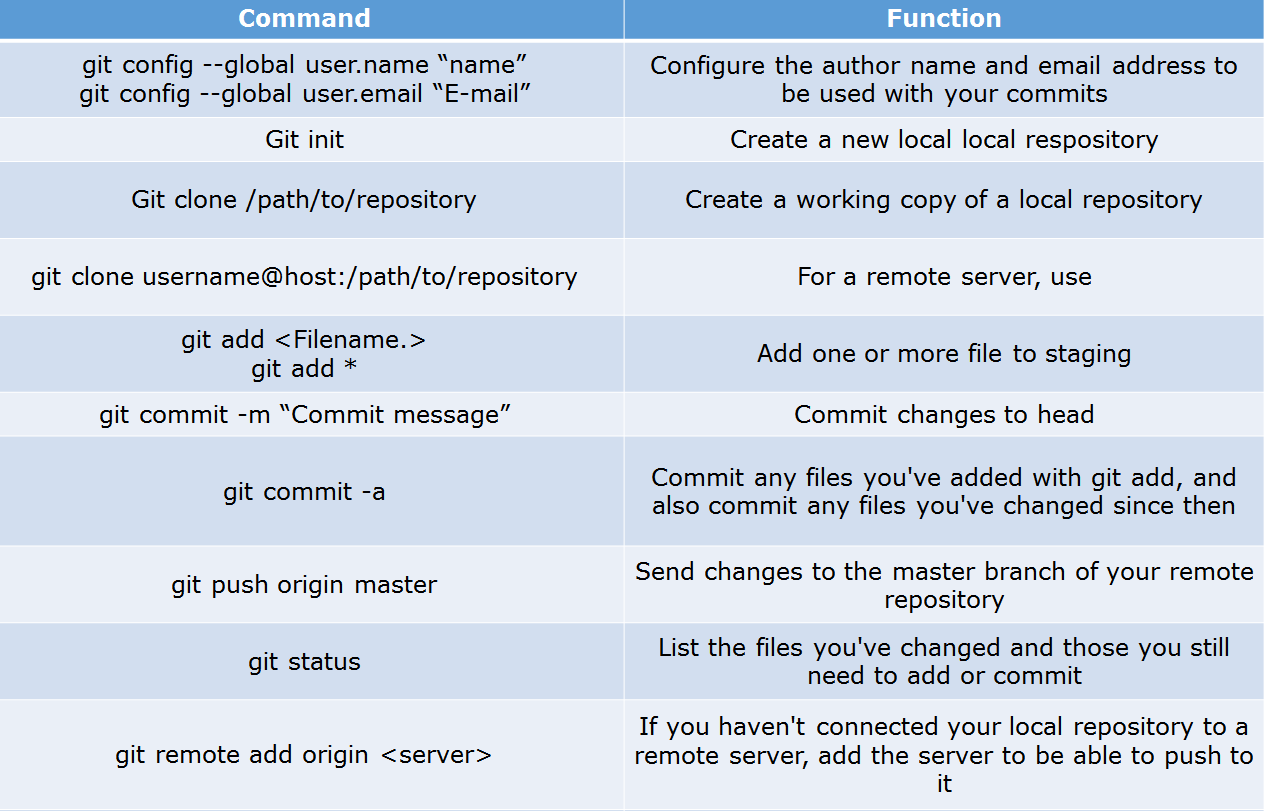
I will suggest that you attempt this question by first explaining about the architecture of git as shown in the below diagram. You can refer to the explanation given below:

* Git is a Distributed Version Control system (DVCS). It can track changes to a file and allows you to revert back to any particular change.
* Its distributed architecture provides many advantages over other Version Control Systems (VCS) like SVN one major advantage is that it does not rely on a central server to store all the versions of a project’s files. Instead, every developer “clones” a copy of a repository I have shown in the diagram below with “Local repository” and has the full history of the project on his hard drive so that when there is a server outage, all you need for recovery is one of your teammate’s local Git repository.
* There is a central cloud repository as well where developers can commit changes and share it with other teammates as you can see in the diagram where all collaborators are committing changes “Remote repository”.



**Explain some basic Git commands?**

Below are some basic Git commands:



**In Git how do you revert a commit that has already been pushed and made public?**

There can be two answers to this question so make sure that you include both because any of the below options can be used depending on the situation:

* Remove or fix the bad file in a new commit and push it to the remote repository. This is the most natural way to fix an error. Once you have made necessary changes to the file, commit it to the remote repository for that I will use  
  **git commit -m “commit message”**
* Create a new commit that undoes all changes that were made in the bad commit.to do this I will use a command  
  **git revert <name of bad commit>**

**How do you squash last N commits into a single commit?**

There are two options to squash last N commits into a single commit. Include both of the below mentioned options in your answer:

* If you want to write the new commit message from scratch use the following command

**git reset –soft HEAD~N &&** **git commit**

* If you want to start editing the new commit message with a concatenation of the existing commit messages then you need to extract those messages and pass them to Git commit for that I will use

**git reset –soft HEAD~N &&** **git commit –edit -m”$(git log –format=%B –reverse .HEAD@{N})”**

**What does Git bisect? How can you use it to determine the source of a (regression) bug?**

I will suggest you to first give a small definition of Git bisect, Git bisect is used to find the commit that introduced a bug by using binary search. Command for Git bisect is  
**git bisect <subcommand> <options>**

Now since you have mentioned the command above, explain what this command will do, this command uses a binary search algorithm to find which commit in your project’s history introduced a bug. You use it by first telling it a “bad” commit that is known to contain the bug, and a “good” commit that is known to be before the bug was introduced. Then Git bisect picks a commit between those two endpoints and asks you whether the selected commit is “good” or “bad”. It continues narrowing down the range until it finds the exact commit that introduced the change.

**What does Git rebase and how can it be used to resolve conflicts in a feature branch before merge?**

According to me, you should start by saying git rebase is a command which will merge another branch into the branch where you are currently working, and move all the local commits that are ahead of the rebased branch to the top of the history on that branch.

Now once you have defined Git rebase time for an example to show how it can be used to resolve conflicts in a feature branch before merge, if a feature branch was created from master, and since then the master branch has received new commits, Git rebase can be used to move the feature branch to the tip of master.

The command effectively will replay the changes made in the feature branch at the tip of master, allowing conflicts to be resolved in the process. When done with care, this will allow the feature branch to be merged into master with relative ease and sometimes as a simple fast-forward operation.

**How do you configure a Git repository to run code sanity checking tools right before making commits, and preventing them if the test fails?**

I will suggest you to first give a small introduction to sanity checking, A sanity or smoke testdetermines whether it is possible and reasonable to continue testing.

Now explain how to achieve this, this can be done with a simple script related to the pre-commit hook of the repository. The pre-commit hook is triggered right before a commit is made, even before you are required to enter a commit message. In this script, one can run other tools, such as linters and perform sanity checks on the changes being committed into the repository.

Finally give an example, you can refer the below script:

**#!/bin/sh  
files=$(git diff –cached –name-only –diff-filter=ACM | grep ‘.go$’)  
if [ -z files ]; then  
exit 0  
fi  
unfmtd=$(gofmt -l $files)  
if [ -z unfmtd ]; then  
exit 0  
fi  
echo “Some .go files are not fmt’d”  
exit 1**

This script checks to see if any .go file that is about to be committed needs to be passed through the standard Go source code formatting tool gofmt. By exiting with a non-zero status, the script effectively prevents the commit from being applied to the repository.

**How do you find a list of files that has changed in a particular commit?**

To get a list files that has changed in a specific commit use command  
**git diff-tree -r {hash}**

Given the commit hash, this will list all the files that were changed or added in that commit. The -r flag makes the command list individual files, rather than collapsing them into root directory names only.

You can also include the below mention point although it is totally optional but will help in impressing the interviewer.

The output will also include some extra information, which can be easily suppressed by including two flags:

**git diff-tree –no-commit-id –name-only -r {hash}**

Here –no-commit-id will suppress the commit hashes from appearing in the output, and –name-only will only print the file names, instead of their paths.

**How do you setup a script to run every time a repository receives new commits through push?**

There are three ways to configure a script to run every time a repository receives new commits through push, one needs to define either a pre-receive, update, or a post-receive hook depending on when exactly the script needs to be triggered.

* Pre-receive hook in the destination repository is invoked when commits are pushed to it. Any script bound to this hook will be executed before any references are updated. This is a useful hook to run scripts that help enforce development policies.
* Update hook works in a similar manner to pre-receive hook, and is also triggered before any updates are actually made. However, the update hook is called once for every commit that has been pushed to the destination repository.
* Finally, post-receive hook in the repository is invoked after the updates have been accepted into the destination repository. This is an ideal place to configure simple deployment scripts, invoke some continuous integration systems, dispatch notification emails to repository maintainers, etc.

Hooks are local to every Git repository and are not versioned. Scripts can either be created within the hooks directory inside the “.git” directory, or they can be created elsewhere and links to those scripts can be placed within the directory.

**How will you know in Git if a branch has already been merged into master?**

I will suggest you include both the below mentioned commands:

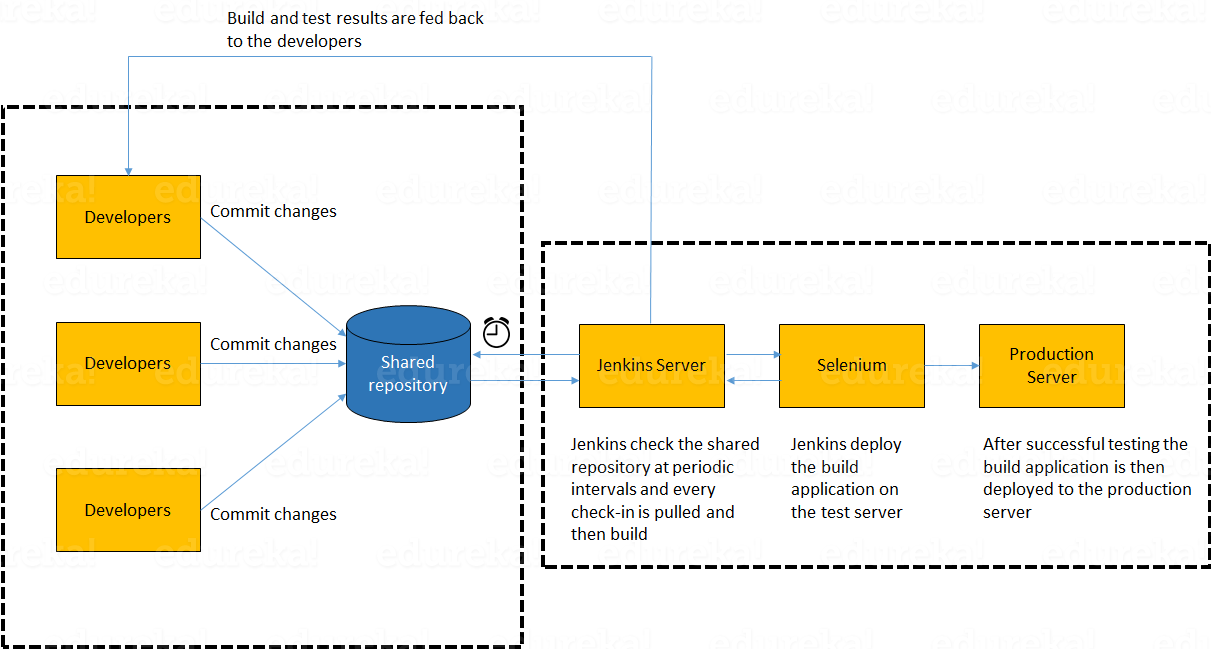
git branch –merged => lists the branches that have been merged into the current branch.  
git branch –no-merged => lists the branches that have not been merged.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Continuous Integration – Jenkins\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**What is meant by Continuous Integration?**

It is a development practice that requires developers to integrate code into a shared repository several times a day. Each check-in is then verified by an automated build, allowing teams to detect problems early.

I suggest that you explain how you have implemented it in your previous job. You can refer the below given example:



In the diagram shown above:

1. Developers check out code into their private workspaces.
2. When they are done with it they commit the changes to the shared repository (Version Control Repository).
3. The CI server monitors the repository and checks out changes when they occur.
4. The CI server then pulls these changes and builds the system and also runs unit and integration tests.
5. The CI server will now inform the team of the successful build.
6. If the build or tests fails, the CI server will alert the team.
7. The team will try to fix the issue at the earliest opportunity.
8. This process keeps on repeating.

**What is Jenkins?**

Jenkins is an open source automation tool written in Java with plugins built for Continuous Integration purpose. Jenkins is used to build and test your software projects continuously making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build. It also allows you to continuously deliver your software by integrating with many testing and deployment technologies.

you can refer the below mentioned use case:

* First, a developer commits the code to the source code repository. Meanwhile, the Jenkins server checks the repository at regular intervals for changes.
* Soon after a commit occurs, the Jenkins server detects the changes that have occurred in the source code repository. Jenkins will pull those changes and will start preparing a new build.
* If the build fails, then the concerned team will be notified.
* If built is successful, then Jenkins deploys the built in the test server.
* After testing, Jenkins generates a feedback and then notifies the developers about the build and test results.
* It will continue to check the source code repository for changes made in the source code and the whole process keeps on repeating.

**What are the benefits of using Jenkins?**

I will suggest you to include the following benefits of Jenkins, if you can recall any other benefit apart from the below mentioned points you can include that as well.

* At integration stage, build failures are cached.
* For each change in the source code an automatic build report notification is generated.
* To notify developers about build report success or failure, it is integrated with LDAP mail server.
* Achieves continuous integration agile development and test-driven development.
* With simple steps, maven release project is automated.
* Easy tracking of bugs at early stage in development environment than production.

**Why do you need a Continuous Integration of Dev & Testing?**

For this answer, you should focus on the need of Continuous Integration. My suggestion would be to mention the below explanation in your answer:

Continuous Integration of Dev and Testing improves the quality of software, and reduces the time taken to deliver it, by replacing the traditional practice of testing after completing all development.

It allows Dev team to easily detect and locate problems early because developers need to integrate code into a shared repository several times a day (more frequently). Each check-in is then automatically tested.

**What are the success factors for Continuous Integration?**

Here you have to mention the requirements for Continuous Integration. You could include the following points in your answer:

* Maintain a code repository
* Automate the build
* Make the build self-testing
* Everyone commits to the baseline every day
* Every commit (to baseline) should be built
* Keep the build fast
* Test in a clone of the production environment
* Make it easy to get the latest deliverables
* Everyone can see the results of the latest build
* Automate deployment

**Explain how you can move or copy Jenkins from one server to another?**

I will approach this task by copying the jobs directory from the old server to the new one.

There are multiple ways to do that:

* Move a job from one installation of Jenkins to another by simply copying the corresponding job directory.
* Make a copy of an existing job by making a clone of a job directory by a different name.
* Rename an existing job by renaming a directory. Note that if you change a job name you will need to change any other job that tries to call the renamed job.

Can you tell me how you deployed? Do u have any deploy plugin?

In recent application team, I have written some shell scripts for deploying into weblogic by providing the credentials. The way it works is from jenkins I have selected the production environment like QA, UAT so based on those parameters it will kick off the scripts. The shell script will have the properties like “envdata” file to get the environment details, like of you select QA then it will get the QA environment details, then it will take the artifacts (like which version to be specified) from nexus repository, then it will copy the artifact to the specified environment and after that it will stop the server and start server and also restarts the server after the deployment is done. Based on that we will get notifications like if the deployment is success or unsuccessful and troubleshoot it if there are any issues.

So this is by using shell scripts, another way to do it is by using cookbooks. I have written cookbooks from the scratch. It also takes the similar architecture but we will install, configure clusters and data sources on the fly for dynamic environments.

**Explain how can create a backup and copy files in Jenkins?**

To create a backup, all you need to do is to periodically back up your JENKINS\_HOME directory. This contains all your build jobs configurations, your slave node configurations, and your build history.

To create a back-up of your Jenkins setup, just copy this directory. You can also copy a job directory to clone or replicate a job or rename the directory.

**Explain how you can setup Jenkins job?**

My approach to this answer will be to first mention how to create Jenkins job. Go to Jenkins top page, select “New Job”, then choose “Build a free-style software project”.  
Then you can tell the elements of this freestyle job:

* Optional SCM, such as CVS or Subversion where your source code resides.
* Optional triggers to control when Jenkins will perform builds.
* Some sort of build script that performs the build (ant, maven, shell script, batch file, etc.) where the real work happens.
* Optional steps to collect information out of the build, such as archiving the artifacts and/or recording javadoc and test results.
* Optional steps to notify other people/systems with the build result, such as sending e-mails, IMs, updating issue tracker, etc.

**Mention some of the useful plugins in Jenkins.**

Below, I have mentioned some important Plugins:

* Maven 2 project
* Amazon EC2
* HTML publisher
* Copy artifact
* Join
* Green Balls

These Plugins, I feel are the most useful plugins. If you want to include any other Plugin that is not mentioned above, you can add them as well. But, make sure you first mention the above stated plugins and then add your own.

**How will you secure Jenkins?**

The way I secure Jenkins is mentioned below. If you have any other way of doing it, please mention it in the comments section below:

* Ensure global security is on.
* Ensure that Jenkins is integrated with my company’s user directory with appropriate plugin.
* Ensure that matrix/Project matrix is enabled to fine tune access.
* Automate the process of setting rights/privileges in Jenkins with custom version controlled script.
* Limit physical access to Jenkins data/folders.
* Periodically run security audits on same.

**Explain how you can deploy a custom build of a core plugin?**

Below are the steps to deploy a custom build of a core plugin:

* Stop Jenkins.
* Copy the custom HPI to **$Jenkins\_Home/plugins**.
* Delete the previously expanded plugin directory.
* Make an empty file called **<plugin>.hpi.pinned**.
* Start Jenkins.

**What you do when you see a broken build for your project in Jenkins?**

There can be multiple answers to this question I will approach this task in the following way:

I will open the console output for the broken build and try to see if any file changes were missed. If I am unable to find the issue that way, then I will clean and update my local workspace to replicate the problem on my local and try to solve it.

**What are the various ways in which build can be scheduled in Jenkins?**

You can schedule a build in Jenkins in the following ways:

* By source code management commits
* After completion of other builds
* Can be scheduled to run at specified time ( crons )
* Manual Build Requests

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Automation Testing\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Different Environment in an organization:**

* Development
* QA == Functional testing of the system
* System Integration Testing == Tests the system from end to end
* SAT = > Site Acceptance Testing => Site Acceptance Testing is the stage where the customer conducts testing for the components supplied under the project scope and tests the conformance of the delivered solution to the Solution Definition Document and functional specifications.
* UAT= > User Acceptance Testing = Allows the user to validate the functionality over time
* Production == Production
* Production Parallel == A parallel of production to replicate production issues
* CCE = Client Certification Environment

**System Acceptance Testing (SAT):**

It is end-to-end testing wherein testing environment is similar to the production environment. We can also called it End to End testing. Here, we navigate through all the features of the software and test if the end business / and feature works. We just test the end feature and don’t check for data flow or do functional testing and all.

**User Acceptance Testing (UAT):**

Acceptance testing is done by end users. Here, they use the s/w for the business for a particular period of time and check whether the s/w can handle all kinds of real-time business scenarios / situations.

**What is Continuous Testing?**

I will advise you to follow the below mentioned explanation:

Continuous Testing is the process of executing automated tests as part of the software delivery pipeline to obtain immediate feedback on the business risks associated with in the latest build.

In this way, each build is tested continuously, allowing Development teams to get fast feedback so that they can prevent those problems from progressing to the next stage of Software delivery life-cycle.

This dramatically speeds up a developer’s workflow as there’s no need to manually rebuild the project and re-run all tests after making changes.

**What is Automation Testing?**

Automation testing or Test Automation is a process of automating the manual process to test the application/system under test. Automation testing involves use of separate testing tools which lets you create test scripts which can be executed repeatedly and doesn’t require any manual intervention.

**What are the benefits of Automation Testing?**

I have listed down some advantages of automation testing. Include these in your answer and you can add your own experience of how Continuous Testing helped your previous company:

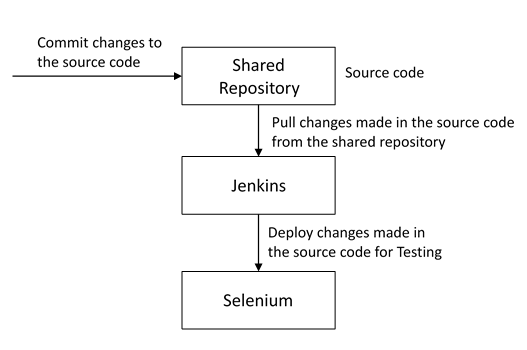
* Supports execution of repeated test cases
* Aids in testing a large test matrix
* Enables parallel execution
* Encourages unattended execution
* Improves accuracy thereby reducing human generated errors
* Saves time and money

**How to automate Testing in DevOps lifecycle?**

I have mentioned a generic flow below which you can refer to:

In DevOps, developers are required to commit all the changes made in the source code to a shared repository. Continuous Integration tools like Jenkins will pull the code from this shared repository every time a change is made in the code and deploy it for Continuous Testing that is done by tools like Selenium as shown in the below diagram.

In this way, any change in the code is continuously tested unlike the traditional approach.



**Why is Continuous Testing important for DevOps?**

You can answer this question by saying, “Continuous Testing allows any change made in the code to be tested immediately. This avoids the problems created by having “big-bang” testing left to the end of the cycle such as release delays and quality issues. In this way, Continuous Testing facilitates more frequent and excellent quality releases.”

**What are the key elements of Continuous Testing tools?**

Key elements of Continuous Testing are:

* **Risk Assessment:** It Covers risk mitigation tasks, technical debt, quality assessment and test coverage optimization to ensure the build is ready to progress toward next stage.
* **Policy Analysis:** It ensures all processes align with the organization’s evolving business and compliance demands are met.
* **Requirements Traceability:** It ensures true requirements are met and rework is not required. An object assessment is used to identify which requirements are at risk, working as expected or require further validation.
* **Advanced Analysis:** It uses automation in areas such as static code analysis, change impact analysis and scope assessment/prioritization to prevent defects in the first place and accomplishing more within each iteration.
* **Test Optimization:** It ensures tests yield accurate outcomes and provide actionable findings. Aspects include Test Data Management, Test Optimization Management and Test Maintenance
* **Service Virtualization:** It ensures access to real-world testing environments. Service visualization enables access to the virtual form of the required testing stages, cutting the waste time to test environment setup and availability.

**Which Testing tool are you comfortable with and what are the benefits of that tool?**

I have mentioned an example below:

I have worked on Selenium to ensure high quality and more frequent releases.

Some advantages of Selenium are:

* It is free and open source
* It has a large user base and helping communities
* It has cross Browser compatibility (Firefox, chrome, Internet Explorer, Safari etc.)
* It has great platform compatibility (Windows, Mac OS, Linux etc.)
* It supports multiple programming languages (Java, C#, Ruby, Python, Pearl etc.)
* It has fresh and regular repository developments
* It supports distributed testing

**What are the Testing types supported by Selenium?**

Selenium supports two types of testing:

**Regression Testing**: It is the act of retesting a product around an area where a bug was fixed.  
**Functional Testing**: It refers to the testing of software features (functional points) individually.

**What is Selenium IDE?**

It is an integrated development environment for Selenium scripts. It is implemented as a Firefox extension, and allows you to record, edit, and debug tests.

Selenium IDE includes the entire Selenium Core, allowing you to easily and quickly record and play back tests in the actual environment that they will run in.

With autocomplete support and the ability to move commands around quickly, Selenium IDE is the ideal environment for creating Selenium tests no matter what style of tests you prefer.

**What is the difference between Assert and Verify commands in Selenium?**

* Assert command checks whether the given condition is true or false. Let’s say we assert whether the given element is present on the web page or not. If the condition is true, then the program control will execute the next test step. But, if the condition is false, the execution would stop and no further test would be executed.
* Verify command also checks whether the given condition is true or false. Irrespective of the condition being true or false, the program execution doesn’t halt i.e. any failure during verification would not stop the execution and all the test steps would be executed.

**How to launch Browser using WebDriver?**

The following syntax can be used to launch Browser:

**WebDriver driver = new FirefoxDriver();**  
**WebDriver driver = new ChromeDriver();**  
**WebDriver driver = new InternetExplorerDriver();**

**When should I use Selenium Grid?**

It can be used to execute same or different test scripts on multiple platforms and browsers concurrently to achieve distributed test execution. This allows testing under different environments and saving execution time remarkably.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Configuration Management \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

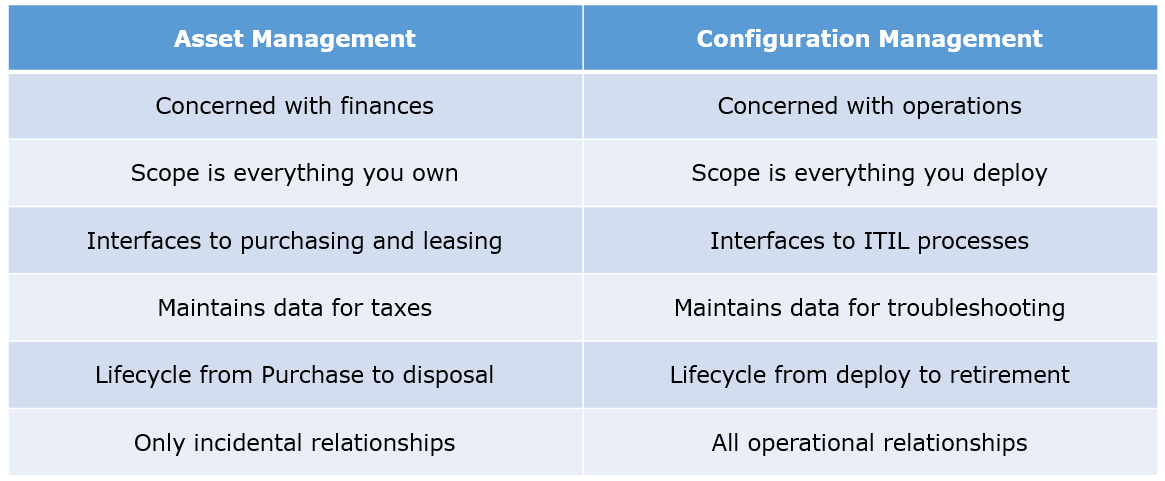
**What are the goals of Configuration management processes?**

The purpose of Configuration Management (CM) is to ensure the integrity of a product or system throughout its life-cycle by making the development or deployment process controllable and repeatable, therefore creating a higher quality product or system. The CM process allows orderly management of system information and system changes for purposes such as to:

* Revise capability,
* Improve performance,
* Reliability or maintainability,
* Extend life,
* Reduce cost,
* Reduce risk and
* Liability, or correct defects.

**Q2. What is the difference between Asset management and Configuration Management?**

Given below are few differences between Asset Management and Configuration Management:



**What is the difference between an Asset and a Configuration Item?**

According to me, you should first explain Asset. It has a financial value along with a depreciation rate attached to it. IT assets are just a sub-set of it.

Anything and everything that has a cost and the organization uses it for its asset value calculation and related benefits in tax calculation falls under Asset Management, and such item is called an asset.

Configuration Item on the other hand may or may not have financial values assigned to it. It will not have any depreciation linked to it. Thus, its life would not be dependent on its financial value but will depend on the time till that item becomes obsolete for the organization.

Now you can give an example that can showcase the similarity and differences between both:

1) Similarity: Server – It is both an asset as well as a CI.  
2) Difference: Building – It is an asset but not a CI.  
3) Document – It is a CI but not an asset

**What do you understand by “Infrastructure as code”? How does it fit into the DevOps methodology? What purpose does it achieve?**

Infrastructure as Code (IAC) is a type of IT infrastructure that operations teams can use to automatically manage and provision through code, rather than using a manual process.  
Companies for faster deployments treat infrastructure like software: as code that can be managed with the DevOps tools and processes. These tools let you make infrastructure changes more easily, rapidly, safely and reliably.

**Which among Puppet, Chef, SaltStack and Ansible is the best Configuration Management (CM) tool? Why?**

This depends on the organization’s need so mention few points on all those tools:

Puppet is the oldest and most mature CM tool. Puppet is a Ruby-based Configuration Management tool, but while it has some free features, much of what makes Puppet great is only available in the paid version.

Organizations that don’t need a lot of extras will find Puppet useful, but those needing more customization will probably need to upgrade to the paid version.

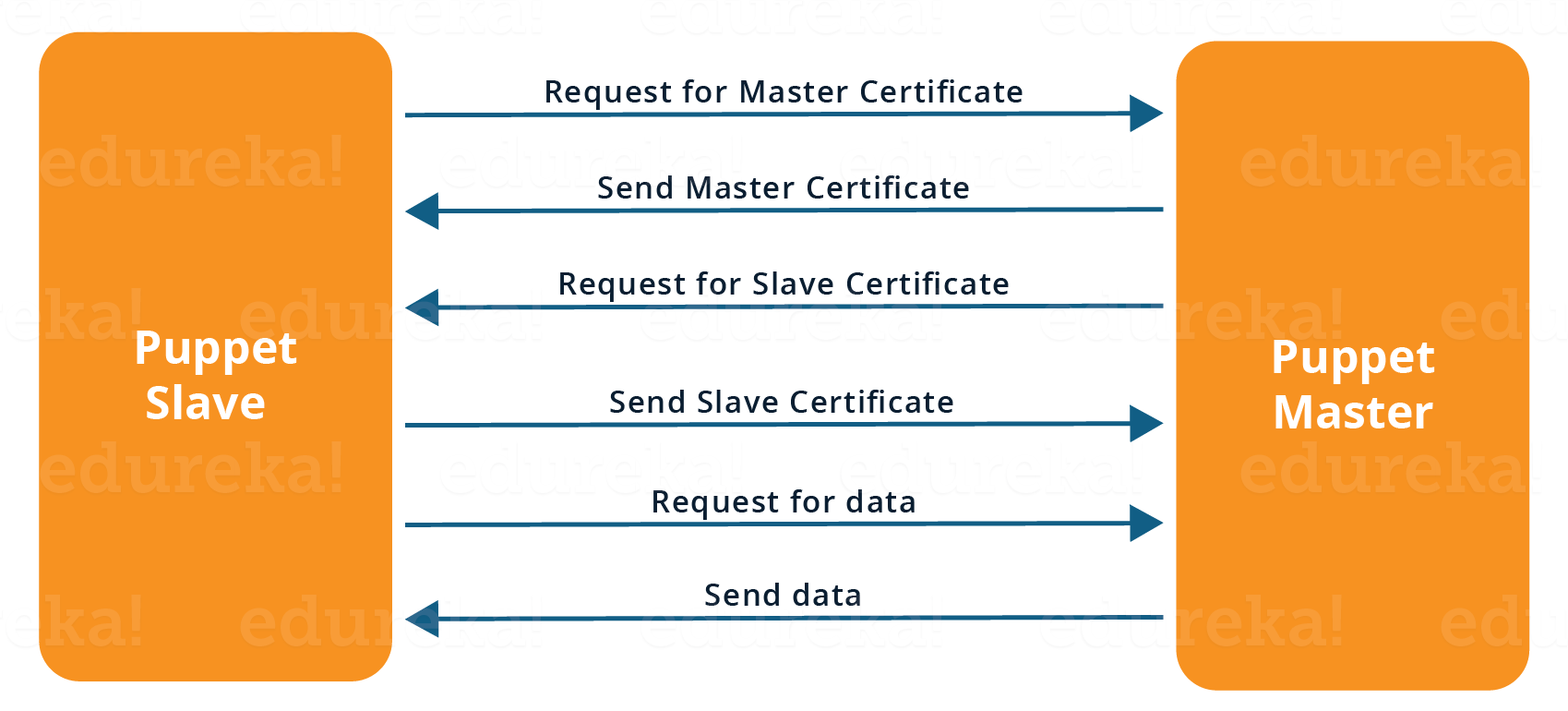
Chef is written in Ruby, so it can be customized by those who know the language. It also includes free features, plus it can be upgraded from open source to enterprise-level if necessary. On top of that, it’s a very flexible product.

Ansible is a very secure option since it uses Secure Shell. It’s a simple tool to use, but it does offer many other services in addition to configuration management. It’s very easy to learn, so it’s perfect for those who don’t have a dedicated IT staff but still need a configuration management tool.

SaltStack is python based open source CM tool made for larger businesses, but its learning curve is low.

**What is Puppet?**

It is a Configuration Management tool which is used to automate administration tasks.  
Now you should describe its architecture and how Puppet manages its Agents. Puppet has a Master-Slave architecture in which the Slave has to first send a Certificate signing request to Master and Master has to sign that Certificate in order to establish a secure connection between Puppet Master and Puppet Slave as shown on the diagram below. Puppet Slave sends request to Puppet Master and Puppet Master then pushes configuration on Slave.  
Refer the diagram below that explains the above description.



**Before a client can authenticate with the Puppet Master, its certs need to be signed and accepted. How will you automate this task?**

The easiest way is to enable auto-signing in puppet.conf.  
Do mention that this is a security risk. If you still want to do this:

* Firewall your puppet master – restrict port tcp/8140 to only networks that you trust.
* Create puppet masters for each ‘trust zone’, and only include the trusted nodes in that Puppet masters manifest.
* Never use a full wildcard such as \*.

**Describe the most significant gain you made from automating a process through Puppet.**

I automated the configuration and deployment of Linux and Windows machines using Puppet. In addition to shortening the processing time from one week to 10 minutes, I used the roles and profiles pattern and documented the purpose of each module in README to ensure that others could update the module using Git. The modules I wrote are still being used, but they’ve been improved by my teammates and members of the community

**Which open source or community tools do you use to make Puppet more powerful?**

Changes and requests are ticketed through Jira and we manage requests through an internal process. Then, we use Git and Puppet’s Code Manager app to manage Puppet code in accordance with best practices. Additionally, we run all of our Puppet changes through our continuous integration pipeline in Jenkins using the beaker testing framework.

**What are Puppet Manifests?**

Every node (or Puppet Agent) has got its configuration details in Puppet Master, written in the native Puppet language.

These details are written in the language which Puppet can understand and are termed as Manifests. They are composed of Puppet code and their filenames use the .pp extension.

You can write a manifest in Puppet Master that creates a file and installs apache on all Puppet Agents (Slaves) connected to the Puppet Master.

**What is Puppet Module and How it is different from Puppet Manifest?**

A Puppet Module is a collection of Manifests and data (such as facts, files, and templates), and they have a specific directory structure. Modules are useful for organizing your Puppet code, because they allow you to split your code into multiple Manifests.

It is considered best practice to use Modules to organize almost all of your Puppet Manifests.  
Puppet programs are called Manifests which are composed of Puppet code and their file names use the .pp extension.

**What is Facter in Puppet?**

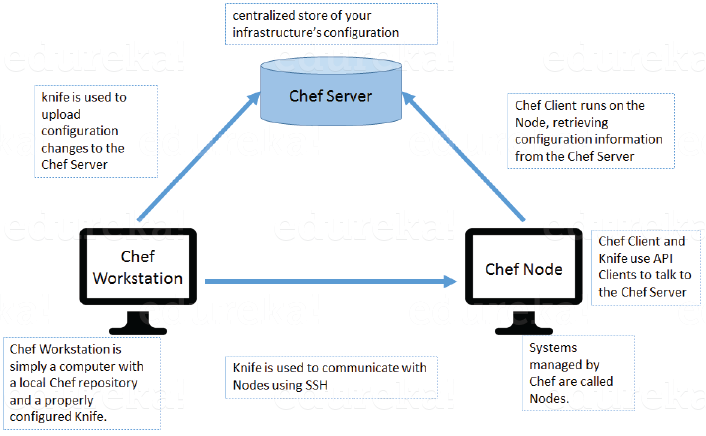
“Facter gathers basic information (facts) about Puppet Agent such as hardware details, network settings, OS type and version, IP addresses, MAC addresses, SSH keys, and more. These facts are then made available in Puppet Master’s Manifests as variables.”

**What is Chef?**

It is a powerful automation platform that transforms infrastructure into code.

Chef is a tool for which you write scripts that are used to automate processes. What processes? Pretty much anything related to IT.

* **Chef Server:**The Chef Server is the central store of your infrastructure’s configuration data. The Chef Server stores the data necessary to configure your nodes and provides search, a powerful tool that allows you to dynamically drive node configuration based on data.
* **Chef Node:** A Node is any host that is configured using Chef-client. Chef-client runs on your nodes, contacting the Chef Server for the information necessary to configure the node. Since a Node is a machine that runs the Chef-client software, nodes are sometimes referred to as “clients”.
* **Chef Workstation:** A Chef Workstation is the host you use to modify your cookbooks and other configuration data.



**What is a resource in Chef?**

A Resource represents a piece of infrastructure and its desired state, such as a package that should be installed, a service that should be running, or a file that should be generated.  
You should explain about the functions of Resource for that include the following points:

* Describes the desired state for a configuration item.
* Declares the steps needed to bring that item to the desired state.
* Specifies a resource type such as package, template, or service.
* Lists additional details (also known as resource properties), as necessary.
* Are grouped into recipes, which describe working configurations.

**What do you mean by recipe in Chef?**

A Recipe is a collection of Resources that describes a particular configuration or policy.

A Recipe describes everything that is required to configure part of a system.  
After the definition, explain the functions of Recipes by including the following points:

* Install and configure software components.
* Manage files.
* Deploy applications.
* Execute other recipes.

**How does a Cookbook differ from a Recipe in Chef?**

“Recipe is a collection of Resources, and primarily configures a software package or some piece of infrastructure. A Cookbook groups together Recipes and other information in a way that is more manageable than having just Recipes alone.”

**What happens when you don’t specify a Resource’s action in Chef?**

when you don’t specify a resource’s action, Chef applies the default action.  
Now explain this with an example, the below resource:  
**file ‘C:\Users\Administrator\chef-repo\settings.ini’ do**  
**content ‘greeting=hello world’**  
**end**  
is same as the below resource:  
**file ‘C:\Users\Administrator\chef-repo\settings.ini’ do**  
**action: create**  
**content ‘greeting=hello world’**  
**end**

**What is Ansible module?**

Modules are considered to be the units of work in Ansible. Each module is mostly standalone and can be written in a standard scripting language such as Python, Perl, Ruby, bash, etc. One of the guiding properties of modules is idempotency, which means that even if an operation is repeated multiple times e.g. upon recovery from an outage, it will always place the system into the same state.

**What are playbooks in Ansible?**

Playbooks are Ansible’s configuration, deployment, and orchestration language. They can describe a policy you want your remote systems to enforce, or a set of steps in a general IT process. Playbooks are designed to be human-readable and are developed in a basic text language.  
At a basic level, playbooks can be used to manage configurations of and deployments to remote machines.

**Q20. How do I see a list of all of the ansible\_ variables?**

Ansible by default gathers “facts” about the machines under management, and these facts can be accessed in Playbooks and in templates.

To see a list of all the facts that are available about a machine, you can run the “setup” module as an ad-hoc action:  
**Ansible -m setup hostname**This will print out a dictionary of all of the facts that are available for that particular host.

**How can I set deployment order for applications?**

WebLogic Server 8.1 allows you to select the load order for applications. See the Application MBean Load Order attribute in Application. WebLogic Server deploys server-level resources (first JDBC and then JMS) before deploying applications. Applications are deployed in this order: connectors, then EJBs, then Web Applications. If the application is an EAR, the individual components are loaded in the order in which they are declared in the application.xml deployment descriptor.

**Can I refresh static components of a deployed application without having to redeploy the entire application?**

Yes, you can use weblogic.Deployer to specify a component and target a server, using the following syntax:  
java weblogic.Deployer -adminurl http://admin:7001 -name appname -targets server1,server2 -deploy jsps/\*.jsp

**How do I turn the auto-deployment feature off?**

The auto-deployment feature checks the applications folder every three seconds to determine whether there are any new applications or any changes to existing applications and then dynamically deploys these changes.

The auto-deployment feature is enabled for servers that run-in development mode. To disable auto-deployment feature, use one of the following methods to place servers in production mode:

* In the Administration Console, click the name of the domain in the left pane, then select the Production Mode checkbox in the right pane.
* At the command line, include the following argument when starting the domain’s Administration Server:  
  -Dweblogic.ProductionModeEnabled=true
* Production mode is set for all WebLogic Server instances in each domain.

**When should I use the \_stage option?**

Set -external\_stage using weblogic.Deployer if you want to stage the application yourself, and prefer to copy it to its target by your own means.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Continuous monitoring\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Why is Continuous monitoring necessary?**

Continuous Monitoring allows timely identification of problems or weaknesses and quick corrective action that helps reduce expenses of an organization. Continuous monitoring provides solution that addresses three operational disciplines known as:

* continuous audit
* continuous controls monitoring
* continuous transaction inspection

**What is Nagios?**

Nagios is one of the monitoring tools. It is used for Continuous monitoring of systems, applications, services, and business processes etc. in a DevOps culture. In the event of a failure, Nagios can alert technical staff of the problem, allowing them to begin remediation processes before outages affect business processes, end-users, or customers. With Nagios, you don’t have to explain why an unseen infrastructure outage affect your organization’s bottom line.

By using Nagios you can:

* Plan for infrastructure upgrades before outdated systems cause failures.
* Respond to issues at the first sign of a problem.
* Automatically fix problems when they are detected.
* Coordinate technical team responses.
* Ensure your organization’s SLAs are being met.
* Ensure IT infrastructure outages have a minimal effect on your organization’s bottom line.
* Monitor your entire infrastructure and business processes.

**How does Nagios works?**

Nagios runs on a server, usually as a daemon or service. Nagios periodically runs plugins residing on the same server, they contact hosts or servers on your network or on the internet. One can view the status information using the web interface.

You can also receive email or SMS notifications if something happens.  
The Nagios daemon behaves like a scheduler that runs certain scripts at certain moments. It stores the results of those scripts and will run other scripts if these results change.

**What are Plugins in Nagios?**

They are scripts (Perl scripts, Shell scripts, etc.) that can run from a command line to check the status of a host or service.

Nagios uses the results from Plugins to determine the current status of hosts and services on your network.

Nagios will execute a Plugin whenever there is a need to check the status of a host or service. Plugin will perform the check and then simply returns the result to Nagios. Nagios will process the results that it receives from the Plugin and take the necessary actions.

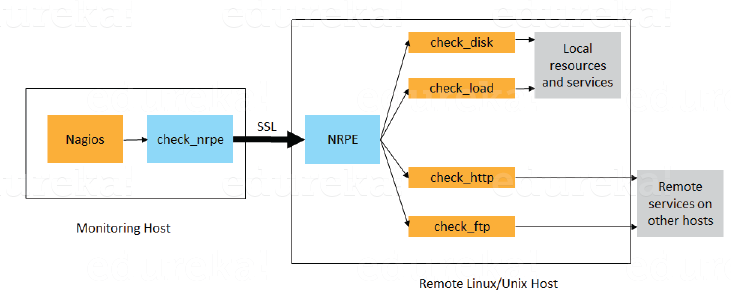
**What is NRPE (Nagios Remote Plugin Executor) in Nagios?**

The NRPE addon is designed to allow you to execute Nagios plugins on remote Linux/Unix machines. The main reason for doing this is to allow Nagios to monitor “local” resources (like CPU load, memory usage, etc.) on remote machines. Since these public resources are not usually exposed to external machines, an agent like NRPE must be installed on the remote Linux/Unix machines.

The NRPE addon consists of two pieces:

* The check\_nrpe plugin, which resides on the local monitoring machine.
* The NRPE daemon, which runs on the remote Linux/Unix machine.

There is a SSL (Secure Socket Layer) connection between monitoring host and remote host as shown in the diagram below.



**What do you mean by passive check in Nagios?**

They are initiated and performed by external applications/processes and the Passive check results are submitted to Nagios for processing.

They are useful for monitoring services that are Asynchronous in nature and cannot be monitored effectively by polling their status on a regularly scheduled basis. They can also be used for monitoring services that are Located behind a firewall and cannot be checked actively from the monitoring host.

**When Does Nagios Check for external commands?**

Nagios check for external commands under the following conditions:

* At regular intervals specified by the command\_check\_interval option in the main configuration file or,
* Immediately after event handlers are executed. This is in addition to the regular cycle of external command checks and is done to provide immediate action if an event handler submits commands to Nagios.

**What is the difference between Active and Passive check in Nagios?**

The major difference between Active and Passive checks is that Active checks are initiated and performed by Nagios, while passive checks are performed by external applications.

Passive checks are useful for monitoring services that are:

* Asynchronous in nature and cannot be monitored effectively by polling their status on a regularly scheduled basis.
* Located behind a firewall and cannot be checked actively from the monitoring host.

The main features of Actives checks are as follows:

* Active checks are initiated by the Nagios process.
* Active checks are run on a regularly scheduled basis.

**How does Nagios help with Distributed Monitoring?**

With Nagios, you can monitor your whole enterprise by using a distributed monitoring scheme in which local slave instances of Nagios perform monitoring tasks and report the results back to a single master. You manage all configuration, notification, and reporting from the master, while the slaves do all the work. This design takes advantage of Nagios’s ability to utilize passive checks i.e. external applications or processes that send results back to Nagios. In a distributed configuration, these external applications are other instances of Nagios.

**Explain Main Configuration file of Nagios and its location?**

The main configuration file contains a number of directives that affect how the Nagios daemon operates.

This config file is read by both the Nagios daemon and the CGIs (It specifies the location of your main configuration file).

Now you can tell where it is present and how it is created. A sample main configuration file is created in the base directory of the Nagios distribution when you run the configure script. The default name of the main configuration file is nagios.cfg. It is usually placed in the etc/ subdirectory of you Nagios installation (i.e. /usr/local/nagios/etc/).

**Explain how Flap Detection works in Nagios?**

Flapping occurs when a service or host changes state too frequently, this causes lot of problem and recovery notifications.

Whenever Nagios checks the status of a host or service, it will check to see if it has started or stopped flapping. Nagios follows the below given procedure to do that:

* Storing the results of the last 21 checks of the host or service analyzing the historical check results and determine where state changes/transitions occur
* Using the state transitions to determine a percent state change value (a measure of change) for the host or service
* Comparing the percent state change value against low and high flapping thresholds

A host or service is determined to have started flapping when it’s percent state change first exceeds a high flapping threshold. A host or service is determined to have stopped flapping when it’s percent state goes below a low flapping threshold.

**What are the three main variables that affect recursion and inheritance in Nagios?**

* Name
* Use
* Register

Name is a placeholder that is used by other objects. Use defines the “parent” object whose properties should be used. Register can have a value of 0 (indicating it’s only a template) and 1 (an actual object). The register value is never inherited.

**What is meant by saying Nagios is Object Oriented?**

One of the features of Nagios is object configuration format in that you can create object definitions that inherit properties from other object definitions and hence the name. This simplifies and clarifies relationships between various components.”

**What is State Stalking in Nagios?**

It is used for logging purposes. When Stalking is enabled for a particular host or service, Nagios will watch that host or service very carefully and log any changes it sees in the output of check results.  
Depending on the discussion between you and interviewer you can also add, “It can be very helpful in later analysis of the log files. Under normal circumstances, the result of a host or service check is only logged if the host or service has changed state since it was last checked.”

**Nagios says my machine is unreachable, not down. What is the difference and how it is achieved?**

When Nagios says a node is unreachable, a node is unreachable if Nagios is not able to find a path to the node.

The node itself may be up but because Nagios is unable to connect to it, it has to mark this as unreachable. To achieve this, Nagios use parent-child relationship between components.

A router may be defined as a parent for a server.

* Now Nagios checks for server and marks it as down.
* It then checks the parent (in our example, the router)
* If parent is also down, then server is marked as unreachable.
* If Parent is up, the server is marked as really down.

**Explain Nagios state types?**

The current state of monitored services and hosts is determined by two components:

* The status of service or host i.e. OK, WARNING, UP, DOWN etc.
* The type of state the service or host is in.

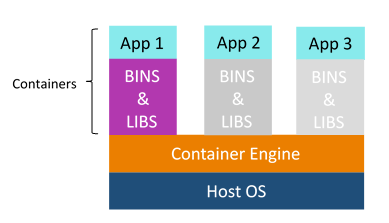
There are two types of states SOFT states and HARD states.

* When a service or host check results are in a non-OK or non-UP state and the service check has not yet been rechecked the number of times specified by the max\_check\_attempts directives in the service or host definition. This is called Soft Error. When a service or a host recovers from Soft Error that is considered as Soft Recovery.
* When a service or host check results are in a non-OK or non-UP state and the service check has been rechecked the number of times specified by the max\_check\_attempts directives in the service or host definition. This is called Hard Error. When a service or a host recovers from Hard Error that is considered as Hard Recovery.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Containerization\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**What are containers?**

containers are used to provide consistent computing environment from a developer’s laptop to a test environment, from a staging environment into production. A container consists of an entire runtime environment: an application, plus all its dependencies, libraries and other binaries, and configuration files needed to run it, bundled into one package. Containerizing the application platform and its dependencies removes the differences in OS distributions and underlying infrastructure.



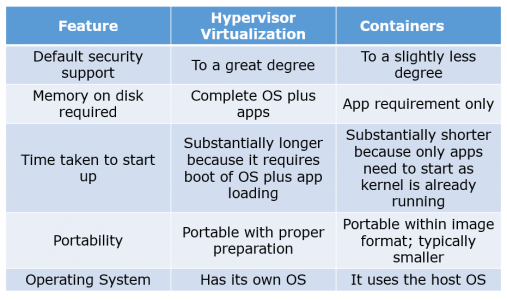
**What are the advantages that Containerization provides over virtualization?**

Below are the advantages of containerization over virtualization:

* Containers provide real-time provisioning and scalability but VMs provide slow provisioning
* Containers are lightweight when compared to VMs
* VMs have limited performance when compared to containers
* Containers have better resource utilization compared to VMs

**How exactly are containers (Docker in our case) different from hypervisor virtualization (vSphere)? What are the benefits?**

Given below are some differences. Make sure you include these differences in your answer:



**What is Docker image?**

Docker image is the source of Docker container. In other words, Docker images are used to create containers. Images are created with the build command, and they’ll produce a container when started with run. Images are stored in a Docker registry such as registry.hub.docker.com because they can become quite large, images are designed to be composed of layers of other images, allowing a minimal amount of data to be sent when transferring images over the network.

**What is Docker container?**

Docker containers include the application and all of its dependencies but share the kernel with other containers, running as isolated processes in user space on the host operating system.

Docker containers are not tied to any specific infrastructure:

They run on any computer, on any infrastructure, and in any cloud.

Docker containers can be created by either creating a Docker image and then running it or you can use Docker images that are present on the Dockerhub.

Docker containers are basically runtime instances of Docker images.

**What is Docker hub?**

Docker hub is a cloud-based registry service which allows you to link to code repositories, build your images and test them, stores manually pushed images, and links to Docker cloud so you can deploy images to your hosts. It provides a centralized resource for container image discovery, distribution and change management, user and team collaboration, and workflow automation throughout the development pipeline.

**How is Docker different from other container technologies?**

Docker containers are easy to deploy in a cloud. It can get more applications running on the same hardware than other technologies, it makes it easy for developers to quickly create, ready-to-run containerized applications and it makes managing and deploying applications much easier.

You can even share containers with your applications.

**What is Docker Swarm?**

It is native clustering for Docker which turns a pool of Docker hosts into a single, virtual Docker host. Docker Swarm serves the standard Docker API, any tool that already communicates with a Docker daemon can use Swarm to transparently scale to multiple hosts.

I will also suggest you to include some supported tools:

* Dokku
* Docker Compose
* Docker Machine
* Jenkins

**What is Dockerfile used for?**

Docker can build images automatically by reading the instructions from a Dockerfile.  
A Dockerfile is a text document that contains all the commands a user could call on the command line to assemble an image. Using docker build users can create an automated build that executes several command-line instructions in succession.

**Can I use json instead of yaml for my compose file in Docker?**

You can use json instead of yaml for your compose file, to use json file with compose, specify the filename to use for ex:  
**docker-compose -f docker-compose.json up**

**Tell us how you have used Docker in your past position?**

Explain how you have used Docker to help rapid deployment. Explain how you have scripted Docker and used Docker with other tools like Puppet, Chef or Jenkins. If you have no past practical experience in Docker and have past experience with other tools in similar space, be honest and explain the same. In this case, it makes sense if you can compare other tools to Docker in terms of functionality.

**How to create Docker container?**

I will suggest you to give a direct answer to this. We can use Docker image to create Docker container by using the below command:

**docker run -t -i <image name> <command name> =>** This command will create and start container.

You should also add, If you want to check the list of all running container with status on a host use the below command:  
**docker ps -a**

**How to stop and restart the Docker container?**

In order to stop the Docker container, you can use the below commands:  
**docker stop <container ID>**  
Now to restart the Docker container you can use:  
**docker restart <container ID>**

**How far do Docker containers scale?**

Large web deployments like Google and Twitter, and platform providers such as Heroku and dotCloud all run on container technology, at a scale of hundreds of thousands or even millions of containers running in parallel.

**What platforms does Docker run on?**

Docker runs on only Linux and Cloud platforms and then I will mention the below vendors of Linux:

* Ubuntu 12.04, 13.04 et al
* Fedora 19/20+
* RHEL 6.5+
* CentOS 6+
* Gentoo
* ArchLinux
* openSUSE 12.3+
* CRUX 3.0+

Cloud:

* Amazon EC2
* Google Compute Engine
* Microsoft Azure
* Rackspace

**Note that Docker does not run on Windows or Mac.**

**Do I lose my data when the Docker container exits?**

No, we won’t lose my data when Docker container exits. Any data that your application writes to disk gets preserved in its container until you explicitly delete the container. The file system for the container persists even after the container halts.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Additional Questions\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**How does HTTP work?**

The HTTP protocol works in a client and server model like most other protocols. A web browser using which a request is initiated is called as a client and a web server software which responds to that request is called a server. World Wide Web Consortium and the Internet Engineering Task Force are two important spokes in the standardization of the HTTP protocol. HTTP allows improvement of its request and response with the help of intermediates, for example a gateway, a proxy, or a tunnel. The resources that can be requested using the HTTP protocol, are made available using a certain type of URI (Uniform Resource Identifier) called a URL (Uniform Resource Locator). TCP (Transmission Control Protocol) is used to establish a connection to the application layer port 80 used by HTTP.

**Explain your understanding and expertise on both the software development side and the technical operations side of an organization you’ve worked for in the past.**

DevOps engineers almost always work in a 24/7 business critical online environment. I was adaptable to on-call duties and able to take up real-time, live-system responsibility. I successfully automated processes to support continuous software deployments. I have experience with public/private clouds, tools like Chef or Puppet, scripting and automation with tools like Python and PHP, and a background in Agile.

**Discuss your experience building bridges between IT Ops, QA and development.**

DevOps is all about effective communication and collaboration. I’ve been able to deal with production issues from the development and operations sides, effectively straddling the two worlds. I’m less interested in finding blame or playing the hero than I am with ensuring that all of the moving parts come together.

**What types of testing are needed?**

Software teams will often look for the “fair weather” path to system completion; that is, they start from an assumption that software will usually work and only occasionally fail. I believe to practice defensive programming in a pragmatic way, which often means assuming that the code will fail and planning for those failures. I try to incorporate unit test strategy, use of test harnesses, early load testing; network simulation, A/B and multi-variate testing  etc.

**Give me an example of how you would handle projects?**

As a professional with managerial responsibilities, I would demonstrate a clear understanding of DevOps project management tactics and also work with teams to set objectives, streamline workflow, maintain scope,  research and introduce new tools or frameworks, translate requirements into workflow and follow up. I would resort to CI, release management and other tools to keep interdisciplinary projects on track.

**What’s your career objective in your role as a DevOps engineer?**

My passion is breaking down the barriers and building and improving processes, so that the engineering and operations teams work better and smarter. That’s why I love DevOps. It’s an opportunity to be involved in the entire delivery system from start to finish.

**How would you make software deployable?**

The ability to script the installation and reconfiguration of software systems is essential towards controlled and automated change. Although there is an increasing trend for new software to enable this, older systems and products suffer from the assumption that changes would be infrequent and minor, and so make automated changes difficult. As a professional who appreciates the need to expose configuration and settings in a manner accessible to automation, I will work with concepts like Inversion of Control (IoC) and Dependency Injection, scripted installation, test harnesses, separation of concerns, command-line tools, and infrastructure as code.

**What is the one most important thing DevOps helps do?**

The most important thing DevOps helps do is to get the changes into production as quickly as possible while minimizing risks in software quality assurance and compliance. That is the primary objective of DevOps. However, there are many other positive side-effects to DevOps. For example, clearer communication and better working relationships between teams which creates a less stressful working environment.

**Which scripting languages do you think are most important for a DevOps engineer?**

As far as scripting languages go, the simpler the better. In fact, the language itself isn’t as important as understanding design patterns and development paradigms such as procedural, object-oriented, or functional programming.

**How do you expect you would be required to multitask as a DevOps professional?**

I believe I’ll be expected to:

1. Focus attention on bridging communication gaps between Development and Operations teams.
2. Understand system design from an architect’s perspective, software development from a developer’s perspective, operations and infrastructure from the perspective of a seasoned Systems Administrator.
3. Execute – to be able to actually do what needs to be done.

**What testing is necessary to ensure that a new service is ready for production?**

DevOps is all about continuous testing throughout the process, starting with development through to production. Everyone shares the testing responsibility. This ensures that developers are delivering code that doesn’t have any errors and is of high quality, and it also helps everyone leverage their time most effectively.

**What’s a PTR in DNS?**

Pointer records are used to map a network interface (IP) to a host name. These are primarily used for reverse DNS. Reverse DNS is setup very similar to how normal (forward) DNS is setup.  When you delegate the DNS forward, the owner of the domain tells the registrar to let your domain use specific name servers.

**Describe two-factor authentication?**

Two-factor authentication is a security process in which the user provides two means of identification from separate categories of credentials; one is typically a physical token, such as a card, and the other is typically something memorized, such as a security code.

**Tell us about the CI tools that you are familiar with?**

The premise of CI is to get feedback as early as possible because the earlier you get feedback, the less things cost to fix. Popular open source tools include Hudson, Jenkins, CruiseControl and CruiseControl.NET. Commercial tools include ThoughtWorks’ Go, Urbancode’s Anthill Pro, Jetbrains’ Team City and Microsoft’s Team Foundation Server.

**What are the advantages of NoSQL database over RDBMS?**

The advantages are:

1. Less need for ETL
2. Support for unstructured text
3. Ability to handle change over  time
4. Breadth of functionality
5. Ability to scale horizontally
6. Support for multiple  data structures
7. Choice of vendors

**What is an MX record in DNS?**

MX records are mail exchange records used for determining the priority of email servers for a domain. The lowest priority email server is the first destination for email. If the lowest priority email server is unavailable, mail will be sent to the higher priority email servers.

**What is the difference between RAID 0 and RAID 1?**

RAID 1 offers redundancy through mirroring, i.e., data is written identically to two drives. RAID 0 offers no redundancy and instead uses striping, i.e., data is split across all the drives. This means RAID 0 offers no fault tolerance; if any of the constituent drives fails, the RAID unit fails.

**How would you prepare for a migration?**

Tips to answer: This question evaluates your experience of real projects with all the awkwardness and complexity they bring. Include terms like cut-over, dress rehearsals, roll-back and roll-forward, DNS solutions, feature toggles, branch by abstraction, and automation in your answer. Developing greenfield systems with little or no existing technology in place is always easier than having to deal with legacy components and configuration. As a candidate if you appreciate that any interesting software system will in effect be under constant migration, you will appear suitable for the role.

**What’s your systems background?**

Tips to answer: Some DevOps jobs require extensive systems knowledge, including server clustering and highly concurrent systems. As a DevOps engineer, you need to analyze system capabilities and implement upgrades for efficiency, scalability and stability, or resilience. It is recommended that you have a solid knowledge of OSes and supporting technologies, like network security, virtual private networks and proxy server configuration.

DevOps relies on virtualization for rapid workload provisioning and allocating compute resources to new VMs to support the next rollout, so it is useful to have in-depth knowledge around popular hypervisors. This should ideally include backup, migration and lifecycle management tactics to protect, optimize and eventually recover computing resources. Some environments may emphasize microservices software development tailored for virtual containers. Operations expertise must include extensive knowledge of systems management tools like Microsoft System Center, Puppet, Nagios and Chef. DevOps jobs with an emphasis on operations require detailed problem-solving, troubleshooting and analytical skills.

**What DevOps tools have you worked with?**

Tips to answer: Software configuration management and build/release (version control) tools, including Apache Subversion, Mercurial, Fossil and others, help document change requests. Developers can more easily follow the company’s best practices and policies while software changes.

Continuous integration (CI) tools such as Rational Build Forge, Jenkins and Semaphore merge all developer copies of the working code into a central version. These tools are important for larger groups where teams of developers work on the same codebase simultaneously. QA experts use code analyzers to test software for bugs, security and performance. If you’ve used HP’s Fortify Static Code Analyzer, talk about how it identified security vulnerabilities in coding languages. Also speak about tools like GrammaTech’s CodeSonar that you used to identify memory leaks, buffer underruns and other defects for C/C++ and Java code. It is essential that you have adequate command of the principal languages like Ruby, C#, .NET, Perl, Python, Java, PHP, Windows PowerShell, and are comfortable with the associated OS environments Windows, Linux and Unix.

**How much have you interacted with cloud based software development?**

Tips to answer: Share your knowledge around use of cloud platforms, provisioning new instances, coding new software iterations with the cloud provider’s APIs or software development kits, configuring clusters to scale computing capacity, managing workload lifecycles and so on. This is the perfect opportunity to discuss container-based cloud instances as an alternative to conventional VMs. Event-based cloud computing, such as AWS Lambda offers another approach to software development, a boon for experienced DevOps candidates. In your interview, mention experience handling big data, which uses highly scalable cloud infrastructures to tackle complex computing tasks.

**What other tools are you familiar with that might help you in this role?**

Tips to answer: DevOps is so diverse and inclusive that it rarely ends with coding, testing and systems. A DevOps project might rely on database platforms like SQL or NoSQL, data structure servers like Redis, or configuration and management issue tracking systems like Redmine. Web applications are popular for modern enterprises, making a background with Web servers, like Microsoft Internet Information Services, Apache Tomcat or other Web servers, beneficial. Make sure to bring across that you are familiar with Agile application lifecycle management techniques and tools.

**Are you familiar with just Linux or have you worked with Windows environments as well?**

Tips to answer: Demonstrate as much as you can, a clear understanding of both the environments including the key tools.

**How can you reduce load time of a dynamic website?**

Tips to answer: Talk about Webpage optimization, cached web pages, quality web hosting , compressed text files, Apache  fine tuning.

**Describe your experience implementing continuous deployment?**

Tips to answer: Answer with a comprehensive list of all the tools that you used. Include inferences of the challenges you faced and how you tackled them.

**How would you ensure traceability?**

Tips to answer: This question probes your attitude to metrics, logging, transaction journeys, and reporting. You should be able to identify that metric, monitoring and logging needs to be a core part of the software system, and that without them, the software is essentially not going to be able to appear maintained and diagnosed. Include words like SysLog, Splunk, error tracking, Nagios, SCOM, Avicode in your answer.

**What was your greatest achievement on a recent project?**

Tips to answer: Make sure you demonstrate your perfect understanding of both development and operations. Do not let your answer lean towards one particular skillset ignoring the other. Even if you have worked in an environment wherein you had to work more with one skillset, assure the interviewer that you are agile according to the needs of your organization.

**What problems did you face and how did you solve them in a way that met the team’s goals?**

Tips to answer: This question aims to find out how much you can handle stress and non-conformity at work. Talk about your leadership skills to handle and motivate the team to solve problems together. Talk about CI, release management and other tools to keep interdisciplinary projects on track.

**Are you more Dev or Ops?**

Tips to answer: This is probably the trickiest question that you might face in the interview. Emphasize the fact that this depends a lot on the job, the company you are working for and the skills of people involved. You really have to be able to alternate between both sides of the fence at any given time. Talk about your experience and demonstrate how you are agile with both.

**What special training or education did it require for you to become a DevOps engineer?**

Tips to answer: DevOps is more of a mind-set or philosophy rather than a skill-set. The typical technical skills associated with DevOps Engineers today is Linux systems administration, scripting, and experience with one of the many continuous integration or configuration management tools like Jenkins and Chef. What it all boils down to is that whatever skill-sets you have, while important, are not as important as having the ability to learn new skills quickly to meet the needs. It’s all about pattern recognition, and having the ability to merge your experiences with current requirements. Proficiency in Windows and Linux systems administration, script development, an understanding of structured programming and object-oriented design, and experience creating and consuming RESTful APIs would take one a long way.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Linux Questions\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

## What is LVM and what is its role?

LVM stands for Logical Volume Manager. It is used to resize the file system online.

## What is the role of Ivextend and Ivreduce?

They are commands used to extend or reduce the volume.

## What is umask?

umask stands for “user file creation mask”. It is used to determine those settings of a mask that control file permission data and directory data.

## In the context of Linux, what does POSIX stand for?

POSIX stands for Portable Operating System Interface for Computer Environments. It is the de facto standard for ensuring compatibility between UNIX versions.

## How is Linux connected to UNIX?

The kernel that Linux uses resembles that of UNIX but does not require UNIX code.

## In the context of Linux, what is GPL?

GPL stands for General Public License. It was originally created to protect the GNU project.

## Explain the GNU project.

GNU is a project that was initiated by Richard Stallman in 1983 at MIT. The project began with the idea of providing control and freedom to the users in using software. As part of the GNU project, users are free to use, run, copy and share software.

## How do you turn a Linux system into a proxy server?

Using the Squid service, you can turn a Linux system into a proxy server.

## What is LILO in the context of Linux?

LILO is a boot loader for Linux. It is used for loading the Linux operating system into the main system memory before beginning operations.

## Define BIOS, MBR and GRUB.

BIOS stands for Basic Input/Output System. MBR is Master Boot Record and GRUB is Graand Unified Bootloader.

## What is the difference between home directory and working directory?

Home directory is the default directory when a user logs in whereas working directory is the user’s current working directory.

## In Linux, how do you track system events?

To track events, a daemon called syslogd is used.

## What do you do when you encounter a suspicious IP?

* We should first block the suspicious IP by integrating tcp\_wrapper.
* Next, we need to enable the “tcp\_wrapper=YES” parameter in the configuration file at ‘/etc/vsftpd.conf’.
* Final step – include the suspicious IP in the ‘host.deny’ file in ‘/etc/host.deny’.

## Describe Telnet and SSH.

Both Telnet and SSH are communication protocols that are used to manage systems remotely. While SSH requires exchange of keys, Telnet transmits data in plain text. Hence, SSH is said to be more secure than Telnet.

## State the difference between the locate and slocate commands.

Locate searches for a file with the latest entries while slocate searches for files that users have accessed most recently.

## How many partitions do you need to install Linux?

You need at least two partitions to install Linux on your system.

## How do you review boot messages in Linux?

By using the dmesg command. Dmesg will pull out boot messages stored in the kernel ring buffer.

## What are symbolic links?

Symbolic links are “shortcut keys” in Linux. These links point to specific programs, files or directories.

## What do Hard Links do?

Hard links point directly to the physical file located on the disk, and not on the path name.

## What does pwd stand for?

In Linux, it stands for print working directory.

## Explain the three different permissions in Linux.

* Read: Gives users the permission to read files or list directories
* Write: Gives users the permission to write to the file of new files and directories
* Execute: Gives users the permission to run the file or lookup a specific file within a directory

## What is the # symbol used for?

# is used for creating new comments.

**What is Linux?**

Linux is an operating system based on UNIX, and was first introduced by Linus Torvalds. It is based on the Linux Kernel, and can run on different hardware platforms manufactured by Intel, MIPS, HP, IBM, SPARC and Motorola. Another popular element in Linux is its mascot, a penguin figure named Tux.

**What is the difference between UNIX and LINUX?**

Unix originally began as a propriety operating system from Bell Laboratories, which later on spawned into different commercial versions. On the other hand, Linux is free, open source and intended as a non-propriety operating system for the masses.

**What is BASH?**

BASH is short for Bourne Again SHell. It was written by Steve Bourne as a replacement to the original Bourne Shell (represented by /bin/sh). It combines all the features from the original version of Bourne Shell, plus additional functions to make it easier and more convenient to use. It has since been adapted as the default shell for most systems running Linux.

**What is Linux Kernel?**

The Linux Kernel is a low-level systems software whose main role is to manage hardware resources for the user. It is also used to provide an interface for user-level interaction.

**What is LILO?**

LILO (Linux Loader) is a boot loader for Linux. It is used mainly to load the Linux operating system into main memory so that it can begin its operations.

**What is a swap space?**

A swap space is a certain amount of space used by Linux to temporarily hold some programs that are running concurrently. This happens when RAM does not have enough memory to hold all programs that are executing.

**What is the advantage of open source?**

Open source allows you to distribute your software, including source codes freely to anyone who is interested. People would then be able to add features and even debug and correct errors that are in the source code. They can even make it run better, and then redistribute these enhanced source code freely again. This eventually benefits everyone in the community.

**What are the basic components of Linux?**

Just like any other typical operating system, Linux has these components:

kernel, shells and GUIs, system utilities, and application program. What makes Linux advantageous over other operating system is that every aspect comes with additional features and all codes for these are downloadable for free.

**Does it help for a Linux system to have multiple desktop environments installed?**

In general, one desktop environment, like KDE or Gnome, is good enough to operate without issues. It’s all a matter of preference for the user, although the system allows switching from one environment to another. Some programs will work on one environment and not work on the other, so it could also be considered a factor in selecting which environment to use.

**What is the basic difference between BASH and DOS?**

The key differences between the BASH and DOS console lies in 3 areas:

* BASH commands are case sensitive while DOS commands are not
* under BASH, / character is a directory separator and \ acts as an escape character. Under DOS, / serves as a command argument delimiter and \ is the directory separator
* DOS follows a convention in naming files, which is 8-character file name followed by a dot and 3 characters for the extension. BASH follows no such convention.

**What is the importance of the GNU project?**

This so-called Free software movement allows several advantages, such as the freedom to run programs for any purpose and freedom to study and modify a program to your needs. It also allows you to redistribute copies of a software to other people, as well as freedom to improve software and have it released to the public.

**Describe the root account.**

The root account is like a systems administrator account, and allows you full control of the system. Here you can create and maintain user accounts, assigning different permissions for each account. It is the default account every time you install Linux.

**What is CLI?**

CLI is short for Command Line Interface. This interface allows user to type declarative commands to instruct the computer to perform operations. CLI offers an advantage in that there is greater flexibility. However, other users who are already accustom with using GUI find it difficult to remember commands including attributes that come with it.

**What is GUI?**

GUI, or Graphical User Interface, makes use of images and icons that users click and manipulate as a way of communicating with the computer. Instead of having to remember and type commands, the use of graphical elements makes it easier to interact with the system, as well as adding more attraction through images, icons and colors.

**How do you open a command prompt when issuing a command?**

To open the default shell (which is where the command prompt can be found), press Ctrl-Alt-F1. This will provide a command line interface (CLI) from which you can run commands as needed.

**How can you find out how much memory Linux is using?**

From a command shell, use the “concatenate” command: cat /proc/meminfo for memory usage information. You should see a line starting something like: Mem: 64655360, etc. This is the total memory Linux thinks it has available to use.

**What is typical size for a swap partition under a Linux system?**

The preferred size for a swap partition is twice the amount of physical memory available on the system. If this is not possible, then the minimum size should be the same as the amount of memory installed.

**What are symbolic links?**

Symbolic links act similarly to shortcuts in Windows. Such links point to programs, files or directories. It also allows you instant access to it without having to go directly to the entire pathname.

**Does the Ctrl+Alt+Del key combination work on Linux?**

Yes, it does. Just like Windows, you can use this key combination to perform a system restart. One difference is that you won’t be getting any confirmation message and therefore, reboot is immediate.

**How do you refer to the parallel port where devices such as printers are connected?**

Whereas under Windows you refer to the parallel port as the LPT port, under Linux you refer to it as /dev/lp . LPT1, LPT2 and LPT3 would therefore be referred to as /dev/lp0, /dev/lp1, or /dev/lp2 under Linux.

**Are drives such as hard drive and floppy drives represented with drive letters?**

No. In Linux, each drive and device has different designations. For example, floppy drives are referred to as /dev/fd0 and /dev/fd1. IDE/EIDE hard drives are referred to as /dev/hda, /dev/hdb, /dev/hdc, and so forth.

**How do you change permissions under Linux?**

Assuming you are the system administrator or the owner of a file or directory, you can grant permission using the chmod command. Use + symbol to add permission or – symbol to deny permission, along with any of the following letters: u (user), g (group), o (others), a (all), r (read), w (write) and x (execute). For example, the command chmod go+rw FILE1.TXT grants read and write access to the file FILE1.TXT, which is assigned to groups and others.

**In Linux, what names are assigned to the different serial ports?**

Serial ports are identified as /dev/ttyS0 to /dev/ttyS7. These are the equivalent names of COM1 to COM8 in Windows.

**How do you access partitions under Linux?**

Linux assigns numbers at the end of the drive identifier. For example, if the first IDE hard drive had three primary partitions, they would be named/numbered, /dev/hda1, /dev/hda2 and /dev/hda3.

**What are hard links?**

Hard links point directly to the physical file on disk, and not on the path name. This means that if you rename or move the original file, the link will not break, since the link is for the file itself, not the path where the file is located.

**What is the maximum length for a filename under Linux?**

Any filename can have a maximum of 255 characters. This limit does not include the path name, so therefore the entire pathname and filename could well exceed 255 characters.

**What are filenames that are preceded by a dot?**

In general, filenames that are preceded by a dot are hidden files. These files can be configuration files that hold important data or setup info. Setting these files as hidden makes it less likely to be accidentally deleted.

**Explain virtual desktop.**

This serves as an alternative to minimizing and maximizing different windows on the current desktop. Using virtual desktops, each desktop is a clean slate where you can open one or more programs. Rather than minimizing/restoring all those programs as needed, you can simply shuffle between virtual desktops with programs intact in each one.

**How do you share a program across different virtual desktops under Linux?**

To share a program across different virtual desktops, in the upper left-hand corner of a program window look for an icon that looks like a pushpin. Pressing this button will “pin” that application in place, making it appear in all virtual desktops, in the same position onscreen.

**What does a nameless (empty) directory represent?**

This empty directory name serves as the nameless base of the Linux file system. This serves as an attachment for all other directories, files, drives and devices.

Daemons are services that provide several functions that may not be available under the base operating system. Its main task is to listen for service request and at the same time to act on these requests. After the service is done, it is then disconnected and waits for further requests.

**How do you switch from one desktop environment to another, such as switching from KDE to Gnome?**

Assuming you have these two environments installed, just log out from the graphical interface. Then at the Log in screen, type your login ID and password and choose which session type you wish to load. This choice will remain your default until you change it to something else.

**How does case sensitivity affect the way you use commands?**

When we talk about case sensitivity, commands are considered identical only if every character is encoded as is, including lowercase and uppercase letters. This means that CD, cd and Cd are three different commands. Entering a command using uppercase letters, where it should be in lowercase, will produce different outputs.

**What are environmental variables?**

Environmental variables are global settings that control the shell’s function as well as that of other Linux programs. Another common term for environmental variables is global shell variables.

**What are the different modes when using vi editor?**

There are 3 modes under vi:

* Command mode – this is the mode where you start in
* Edit mode – this is the mode that allows you to do text editing
* Ex mode – this is the mode wherein you interact with vi with instructions to process a file

**Is it possible to use shortcut for a long pathname?**

Yes, there is. A feature known as filename expansion allows you to do this using the TAB key. For example, if you have a path named /home/iceman/assignments directory, you would type as follows: /ho[tab]/ice[tab]/assi[tab]. This, however, assumes that the path is unique, and that the shell you’re using supports this feature.

**What is redirection?**

Redirection is the process of directing data from one output to another. It can also be used to direct an output as an input to another process.

**What is grep command?**

grep a search command that makes use of pattern-based searching. It makes use of options and parameters that is specified along the command line and applies this pattern into searching the required file output.

**What could possibly be the problem when a command that was issued gave a different result from the last time it was used?**

One highly possible reason for getting different results from what seems to be the same command has something to do with case sensitivity issues. Since Linux is case sensitive, a command that was previously used might have been entered in a different format from the present one. For example, to lists all files in the directory, you should type the command ls, and not LS. Typing LS would either result in an error message if there is no program by that exact name exist, or may produce a different output if there is a program named LS that performs another function.

**What are the contents in /usr/local?**

It contains locally installed files. This directory actually matters in environments where files are stored on the network. Specifically, locally-installed files go to /usr/local/bin, /usr/local/lib, etc.). Another application of this directory is that it is used for software packages installed from source, or software not officially shipped with the distribution.

**How do you terminate an ongoing process?**

Every process in the system is identified by a unique process id or pid. Use the kill command followed by the pid in order to terminate that process. To terminate all process at once, use kill 0.

**How do you insert comments in the command line prompt?**

Comments are created by typing the # symbol before the actual comment text. This tells the shell to completely ignore what follows.

For example: “# This is just a comment that the shell will ignore.”

**What is command grouping and how does it work?**

You can use parentheses to group commands. For example, if you want to send the current date and time along with the contents of a file named OUTPUT to a second file named MYDATES, you can apply command grouping as follows: (date cat OUTPUT) > MYDATES

**How do you execute more than one command or program from a single command line entry?**

You can combine several commands by separating each command or program using a semicolon symbol. For example, you can issue such a series of commands in a single entry:

|  |
| --- |
| ls –l cd .. ls –a MYWORK which is equivalent to 3 commands: ls -l cd.. ls -a MYWORK |

\*\*Note that this will be executed one after the other, in the order specified.

**Write a command that will look for files with an extension “c”, and has the occurrence of the string “apple” in it.**

Find ./ -name “\*.c” | xargs grep –i “apple”

**Write a command that will display all .txt files, including its individual permission.**

|  |
| --- |
| ls -a -l \*.txt |

**Write a command that will do the following:  
-look for all files in the current and subsequent directories with an extension c,v  
-strip the,v from the result (you can use sed command)  
-use the result and use a grep command to search for all occurrences of the word ORANGE in the files.**

|  |
| --- |
| Find ./ -name “\*.c,v” | sed ‘s/,v//g’ | xargs grep “ORANGE” |

**What, if anything, is wrong with each of the following commands?  
a) ls -l-s  
b) cat file1, file2  
c) ls – s Factdir**

Answers:  
a) there should be space between the 2 options: ls -l -s  
b) do not use commas to separate arguments: cat file1 file2  
c) there should be no space between hyphen and option label: ls –s Factdir

**What is the command to calculate the size of a folder?**

To calculate the size of a folder use the command **du –sh folder1.**

**How can you find status of a process?**

Use the command

ps ux

**How can you check the memory status ?**

You can use the command

free -m  to display output in MB

free -g  to display output in GB

**Explain how to color the Git console?**

To color the Git console, you can use the command **git config—global color.ui auto.** In the command, the color.ui variable sets the default value for variable such as **color.diff** and **color.grep.**

**How can you append one file to another in  Linux?**

To append one file to another in Linux you can use command **cat file2 >> file 1.** The operator >> appends the output of the named file or creates the file if it is not created.  While another command **cat file 1 file 2 > file 3** appends two or more files to one.

**Explain how you can find a file using Terminal?**

To find a file you have to use command, **find . –name “process.txt”** .  It will look for the current directory for a file called process.txt.

**Explain how you can create a folder using Terminal?**

To create a folder, you have to use command ***mkdir.*** It will be something like these :  ~$ mkdir Guru99

**Explain how you can view the text file using Terminal?**

To view the text file, go to the specific folder where the text files are located by using the command **cd** and then type **less filename.txt.**

**Explain how to enable curl on Ubuntu LAMP stack?**

To enable curl on Ubuntu , first install libcurl, once done use following command **sudo/etc/init .d /apache2 restart** or **sudo service apache2 restart.**

**Explain how to enable root loging in Ubuntu?**

The command which enables root loging is

#sudo sh-c ‘echo “greater-show-manual-login=true” >>/etc/lightdm/lightdm.conf’

**How you can run an Linux program in the background simultaneously when you start your Linux Server?**

By using **nohup.**It will stop the process receiving the **NOHUP** signal and thus terminating it you log out of the program which was invoked with.  **&** runs the process in the background.

**Explain how to uninstall the libraries in Linux?**

To uninstall the libraries in Linux, you can use command

sudo apt – get remove library\_name

What happens at the boot up process?

* When you turn on the power, the power is distributed to all parts of the system.
* POST – Power on Self-Test
* BIOS initializes tests to make sure devices connected are working all right like printer, keyboard, mouse, speakers
* The third step is BIOS check the hard drive and Boot disk / OS. Once it is confirmed that there is OS, This OS is loaded into RAM. When this process is done we will see the screen.

**Other References:**

https://github.com/spikenode/DevOps-Interview-Questions#general-questions